



Pepperl+Fuchs[®], Inc. • 1600 Enterprise Parkway • Twinsburg, OH 44087
Telephone (330) 425-3555 • FAX (330) 425-4607

Mini Memory Monitor

Custom Application

for

The Pepperl+Fuchs Microwave Identification System Communicator

MTT-S1

Version 3.2

Subject: P+F Mini-Mem Monitor for the MTT-S1 Communicator

The Custom Application is divided into several modules that perform different operations.

1) “Set-Up” Module

When the Communicator is powered up the Custom Application performs an internal check of its hardware components. After the internal hardware check is complete, the Custom Application is in the *set-up mode*. While the system is in the *set-up mode*, the main LED will blink amber and the Communicator attempts to read a *set-up tag*. The set-up tag contains all of the necessary parameters concerning the serial communications and the microwave link between the tag and the Communicator. This set-up tag **must be formatted to Full-Memory** or the Custom Application will reject the set-up data on the tag (main LED blinks red). After the Communicator reads a set-up tag the appropriate parameters are changed and the Communicator starts to read tags immediately. Also, when the communicator is powered up and the S3 push button is pressed immediately the default configuration settings are stored.

If the Communicator does not read a *set-up tag* it will remain in the *set-up mode* for 10 seconds. After this ten seconds delay time the Communicator reports the system set-up information in the following format:

P+F Mini-Mem Monitor VER 3.2: November-20-2003 by Pepperl+Fuchs, Inc.

After this initial status report the Communicator starts reading tags.

1.1) The Read Module

After exiting the *Set-Up Module*, the Custom Application enters the *Read Module*. This is the normal operation since the Communicator reads tags and sends information to serial ports A and B. There are four different modes that mini memory monitor can be operated in which are as follows: standard, enhanced, track and universal mode. Standard is the basic operating mode while enhanced mode is an expansion of the standard, and track mode is an expansion of enhanced. Universal mode is a significant modification enabling additional interfacing between the Communicator and the control PC. The different operating modes modify the way data is read from the data carriers and transmitted to the serial ports.

1.3) Electrical Connections

This Custom Application supports RS232 communications via connector J2, RS485 and RS422 via connector J3. Furthermore, Input 1, Input 2, Input 3 and Output 1 are supported.

J1: LED Outputs

Two LED-outputs are used as an *alive* or *health* indicator. Once the Custom Application is executing the *Read-Module* the two LED outputs “blink” once/second. If desired, the two *alive* signals can be read by a PLC.

Pin 1 is the LED output 1

Pin 2 is the LED output 2

Pin 3 is the LED GND

J2: Programming Port/Communications (RS232 port-a)

The programming port (RS232 port-a) is serial communications port and is dedicated for RS232 serial communications. A custom Application software must be downloaded to the MTT-S1 via this port.

Pin 1 is the RS232 transmit line of the Communicator. If the Communicator is connected to a standard DB-9 port on a PC this pin must be connected to pin 2 on the PC side.

Pin 2 is the RS232 receive line of the Communicator. If the Communicator is connected to a standard DB-9 port on a PC this pin must be connected to pin 3 on the PC side.

Pin 3 is the RS232 Ground. If the Communicator is connected to a standard DB-9 port on a PC this pin must be connected to pin 5 on the PC side.

J3: Communications (RS232 port-b, RS485/RS422 port)

Communication port-b can be configured for RS232, RS485 and RS422. The following data will be sent over this port:

In normal and track mode the data on this port is identical to that on RS232 port-a.

In enhanced mode this port is used to send the emulated barcode data.

In universal mode this port sends emulated barcode data.

Communications options for Universal Mode (RS232 and RS485)

Universal mode supports RS232, RS422, and RS485 communications. For RS485 and RS422 the jumper located directly under the main LED D1 must be in the 485 position. It is also necessary to configure the Communicator firmware using a configuration tag or the push buttons. Data received on either port (A or B) are treated identical.

RS232 port b

The dual-jumper under the main LED D1 must be in the 232 position.

Pin 1 is the RS232 transmit line of the Communicator. If the Communicator is connected to a standard DB-9 port on a PC this pin must be connected to pin 2 on the PC side.

Pin 2 is the RS232 receive line of the Communicator. If the Communicator is connected to a standard DB-9 port on a PC this pin must be connected to pin 3 on the PC side.

Pin 3 is the RS232 Ground. If the Communicator is connected to a standard DB-9 port on a PC this pin must be connected to pin 5 on the PC side.

RS422

The dual-jumper under the main LED D1 must be in the 485 position.

Pin 5 is the RS422 transmit (Tx-) line of the Communicator. If the communicator is connected to another device via RS422 this pin must be connected to receive (Rx-) line of that device.

Pin 6 is the RS422 transmit (Tx+) line of the Communicator. If the communicator is connected to another device via RS422 this pin must be connected to receive (Rx+) line of that device.

Pin 7 is the RS422 receive (Rx-) line of the Communicator. If the communicator is connected to another device via RS422 this pin must be connected to transmit (Tx-) line of that device.

Pin 8 is the RS422 receive (Rx+) line of the Communicator. If the communicator is connected to another device via RS422 this pin must be connected to transmit (Tx+) line of that device.

RS485

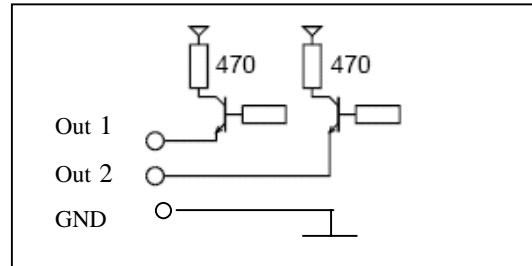
The dual-jumper under the main LED D1 must be in the 485 position.

Pin 5 is the RS485 transmit/receive (Tx/Rx-) line of the Communicator. If the communicator is connected to another device via RS485 this pin must be connected to transmit (Tx-) line of that device.

Pin 6 is the RS485 transmit/receive (Tx/Rx+) line of the Communicator. If the communicator is connected to another device via RS485 this pin must be connected to transmit (Tx+) line of that device.

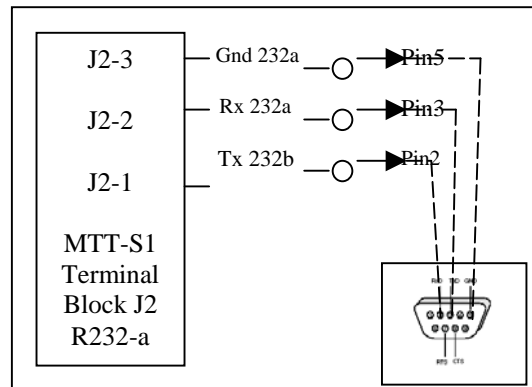
J1: LED'S

Pin 1 LED output 1
Pin 2 LED output 2
Pin 3 LED GND



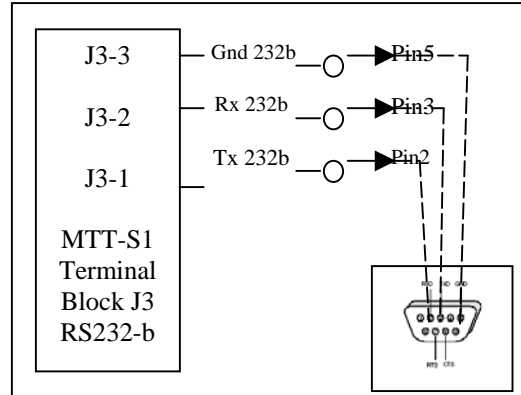
J2: RS232 port-a via pins 1, 2, and 3

Pin 1 RS232a transmit line
Pin 2 RS232a receive line
Pin 3 RS232a ground



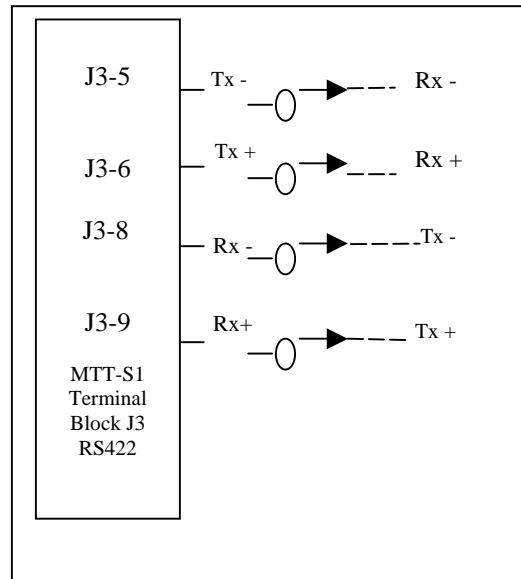
J3: RS232 Port-b port via pins 1,2,3

- Pin 1 RS232b transmit line
- Pin 2 RS232b receive line
- Pin 3 RS232b ground



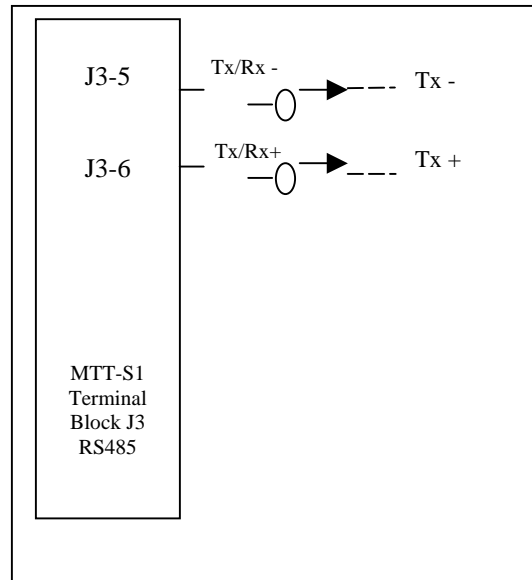
RS422 port via pins 5, 6, 8 and 9

- Pin 5 RS422 Transmit (Tx -)
- Pin 6 RS422 Transmit (Tx +)
- Pin 8 RS422 Receive (Rx -)
- Pin 9 RS422 Receive (Rx +)



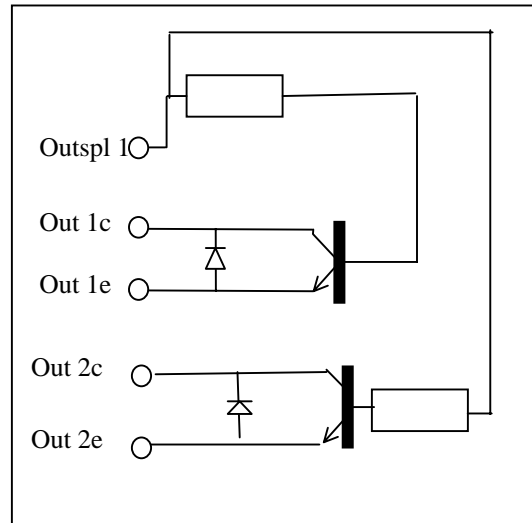
RS485 port via pins 5, and 6

Pin 5 RS485 Transmit/Receive (Tx/Rx -)
Pin 6 RS485 Transmit/Receive (Tx/Rx +)



J4: Output via pins 1, 2 and 3

Pin 1 Output Supply
Pin 2 (Out 1c)
Pin 3 (Out 1e)
Pin 4 (Out 2c)
Pin 5 (Out 2e)



J5: Input1 via pins 1 and 2

Pin 1 (In 1a)

Pin 2 (In 1c)

A 24 V input signal can be connected between In 1a and In 1c

Input2 via pins 3 and 4

Pin 3 (In 2a)

Pin 4 (In 2c)

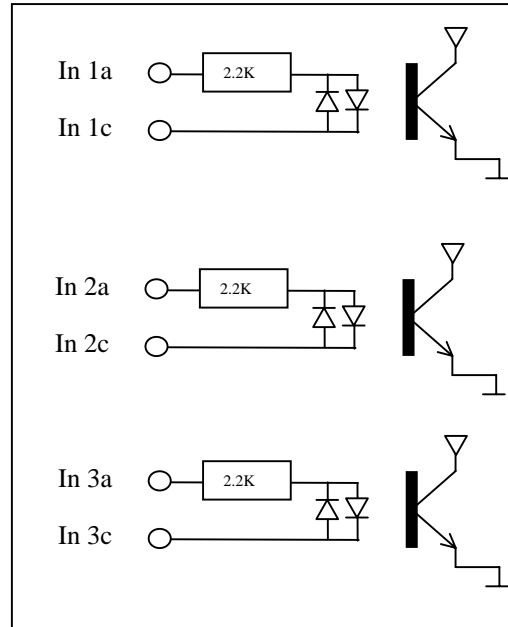
A 24 V input signal can be connected between In 2a and In 2c

Input3 via pins 5 and 6

Pin 5 (In 3a)

Pin 6 (In 3c)

A 24 V input signal can be connected between In 3a and In 3c



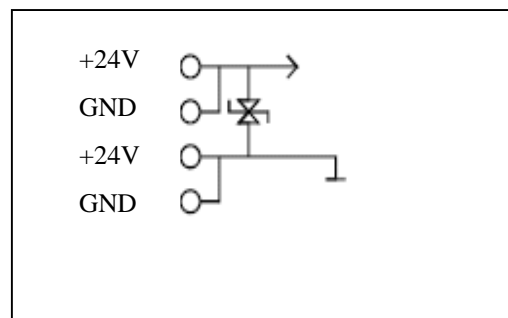
J6: 24 VDC Power Supply

Pin 1 (Spl 1) +24 V

Pin 2 (Rtnspl 1) GND

Pin 3 (Spl 2) +24 V

Pin 4 (Rtnspl 2) GND



1.4) Push Button Configuration

The internal push buttons S2 and S3 can be used to check and set the configuration of the Communicator. The push button S2 is used to toggle through the available configuration options. The push button S3 is used to toggle through the available parameters. Once a parameter has been selected, pushing S2 acknowledges the new setting and returns to the configuration option. A short beep indicates that the setting has been stored.

**Please note when configuration options are changed the power of the communicator must be reset.

The following configuration and parameter settings are available:

Display Characters	Parameter Value	Scope	Description
OP	1		Standard mode
	2		Enhanced mode
	3		Track mode
	4		Universal mode
cF	01 to 99	All modes *(Default value of microwave channel = 50)	Microwave Channel
cr	1 to 4	All modes *(Default value of range setting= 1)	Range setting
br	0	All Modes	4800 baud
	1		9600 baud
	2		19200 baud
Pb	23	All Modes	RS232
	48	Standard, track, and universal mode	RS485
	42	Standard, track, and universal mode	RS422
FF	F	Enhanced, track and universal mode	Fixed length reply
	U		Variable length reply
Fn	1 to 5	Enhanced, track and universal mode	Length of reply
Ft	00 to FF	Enhanced, track and universal mode *(Default value of footer= <CR>)	Hex value of terminator
Fh	00 to FF	Track and universal mode *(Default value of header= 21 Hex)	Hex value of header
FS	00 to FF	Track mode *(Default value of handshake= NONE)	Hex value of handshake
hb	0	Universal mode	Heartbeat disabled
	1		Unsolicited heartbeat (1min)
	2		Heartbeat with Handshake
Sc	NO	All Modes	DO NOT write configuration
	DO		Write configuration to tag
rn	0	Track mode	<i>Read Always</i> , LON and LOFF are ignored
	1 to 9		Number of reads

2) Standard Mode Operation

The standard mode of operation is used to read and verify tag data and report the 16-bit data string to both the RS232 port-a and port-b.

The following push button configuration and parameter settings are available in standard mode below:

Display Characters	Parameter Value	Scope	Description
OP	1		Standard mode
cF	01 to 99	All modes *(Default value of microwave channel = 50)	Microwave Channel
cr	1 to 4	All modes *(Default value of range setting= 1)	Range setting
br	0	All Modes	4800 baud
	1		9600 baud
	2		19200 baud
Pb	23	All Modes	RS232
	48	Standard, track, and universal mode	RS485
	42	Standard, track, and universal mode	RS422
Sc	NO	All Modes	DO NOT write configuration
	DO		Write configuration to tag

2.1) Format of the Set-Up Tag for Standard Mode

After power up, the Custom Application is looking for tags in the read zone. Once a tag has been read, the data from that string is evaluated and tested for validity. A valid set-up tag must be formatted as **Full-Memory**. This procedure assures that only true set up tags are used. All tags that do not contain data compatible with the format of the set-up tags are rejected. A valid set-up tag must have the following format: *(Set up tags only read on power-up)

\$\$\$\$S(text,5)C(number,2)R(number,1)B(number,1)

- \$\$\$\$ This is a user definable string that precedes the set-up data. Its sole purpose is to make the data string unique to that communicator.
- S(text,5) The letter "S" followed by exactly 5 characters is used as a station ID and is stored in memory. Currently, the station ID is not used at other places in the Custom Application.
- C(number,2) The letter "C" followed by exactly 2 digits defines the microwave channel used for communications. The range of possible microwave channels ranges from **01** to **99**. This, being one of the most powerful features of the Ident-M System T Communicators, allows them to be placed in close proximity to one another without the chance of interference.
- R(number,1) The letter "R" followed by exactly 1 digit sets the read range of the Communicator. Numbers between **1** and **4** are possible, where 1 defines the shortest read range and 4 sets the longest read range.

B(number,1) The letter “B” followed by exactly 1 digit sets the baud rate and mode of serial communications. The possible numbers are:

0 for 4800 baud, RS232	5 for 4800, RS485	8 for 4800, RS422
1 for 9600 baud, RS232	6 for 9600, RS485	9 for 9600, RS422
2 for 19200 baud, RS232	7 for 19200, RS485	A for 19200, RS422

2.2) Reading and Reporting of Tags

After power up, the Custom Application will read all tags as fast as possible. If a tag enters the read zone, it is read immediately. After a 500 ms “dead time” the tag is read again.

After a tag has been read and verified (only tags formatted to **Mini-Memory** are allowed) it is immediately reported via the serial interface. Both serial interfaces (port-a and port-b) will show the same data. If the tag remains in the read zone it is not reported again. If a tag leaves the read zone for more than 1s it is considered a new tag and reported again via the serial interface. A new tag is always reported immediately, even if it enters the communications zone a few milliseconds after a previously read tag.

In the data handling, 8 bits are sent in the first byte and the remaining 6 bits are sent in the second Byte. Bit 7 of the second byte is enabled for low battery and bit 8 is used for the “g” command.

a.) Digital Outputs

The digital outputs on the Communicator will be used to indicate that tag data is available on the serial interface. These digital outputs will remain high for 100 ms after the data word was sent out. In addition to those digital outputs, two so-called LED-outputs are used as an *alive* or *health* indicator. After power up the two LED outputs “blink” once/second. If desired, the two *alive* signals can be read by a PLC.

b.) Digital Input

Digital input 1 on the Communicator is used to request the last data from the Communicator. Once the Custom Application receives a *high* on this input the most recent data word is sent to the serial interface again.

Digital input 2 on the Communicator is used to request a re-read. If a tag is in the field when the digital input 2 is selected, the data on that tag will be reported. Instead the result of this operation reflects the current state of the system. If no tag is in the field the *No Tag* reply (bf_{hex}) will be issued after 1 second.

c.) g<cr> - Command

Data can also be requested via the g-command. When this command is issued before the Communicator reads any tag it will respond with bf_{hex}.

To issue this command the Communicator must receive g<cr> equal to 67_{hex}0D_{hex}. This command can be issued via the serial ports a or b.

The function of this command is identical to setting input 1, always old data, Bit 15=1.

d.) n<cr> - Command

Data can also be requested via the n-command. The function of this command is identical to setting input 2. The new data bit is always low.

To issue this command the communicator must receive n<cr> equal to 6E_{hex}0D_{hex}. This command can be issued via the serial port-a, and port-b.

e.) s<cr> - Command

Data can also be requested via the s-command. The function of this command is that it will return the string of data sent on boot up.

To issue this command the communicator must receive s<cr> equal to 73_{hex}0D_{hex}. This command can be issued via the serial port-a, and port-b.

f.) f<cr> - Command

Data can also be requested via the f-command. The function of this command is that it returns the same information as the g<cr> except the two-character tag format is added to the end. The tag format can be one of the following: **C0, C4, C8, C6, R4, R8, R6**

To issue this command the communicator must receive f<cr> equal to 66_{hex}0D_{hex}. This command can be issued via serial port-a, and port-b.

3) Enhanced Mode Operation

The enhanced mode of operation is an extension of the standard mode and is used to read tag data and report the 2 byte data string to RS232 port-a as it does in standard mode. This mode also allows the sending of emulated barcode data including postamble characters to the serial port-b if a trigger occurs while the tag is in the read zone.

The following push button configuration and parameter settings are available in enhanced mode below:

Display Characters	Parameter Value	Scope	Description
OP	2		Enhanced mode
cF	01 to 99	All modes *(Default value of microwave channel = 50)	Microwave Channel
cr	1 to 4	All modes *(Default value of range setting= 1)	Range setting
br	0	All Modes	4800 baud
	1		9600 baud
	2		19200 baud
Pb	23	All Modes	RS232
	48	Standard, track, and universal mode	RS485
	42	Standard, track, and universal mode	RS422
FF	F	Enhanced, track and universal mode	Fixed length reply
	U		Variable length reply
Fn	1 to 5	Enhanced, track and universal mode	Length of reply
Ft	00 to FF	Enhanced, track and universal mode *(Default value of footer=< CR >)	Hex value of terminator
Sc	NO	All Modes	DO NOT write configuration
	DO		Write configuration to tag

3.1) Format of the Set-Up Tag for Enhanced Mode

On power up, the Custom Application is looking for tags in the read zone. Once a tag has been read, the data from that string is evaluated and tested for validity. A valid set-up tag must be formatted as **Full-Memory**. This procedure assures that only true set up tags are used. All tags that do not contain data compatible with the format of the set-up tags are rejected. A valid set-up tag must have the following format: ******(Set up tags only read on power-up)

\$\$\$\$S(text,5)C(number,2)R(number,1)B(number,1)L(xn,2)T(ASCII,2)

- \$\$\$\$ This is a user definable string that precedes the set-up data. Its sole purpose is to make the data string unique to that communicator.
- S(text,5) The letter "S" followed by exactly 5 characters is used as a station ID and is stored in memory. Currently, the station ID is not used at other places in the Custom Application.
- C(number,2) The letter "C" followed by exactly 2 digits defines the microwave channel used for communications. The range of possible microwave channels ranges from **01** to **99**. This,

being one of the most powerful features of the Ident-M System T Communicators, allows them to be placed in close proximity to one another without the chance of interference.

R(number,1) The letter “R” followed by exactly 1 digit sets the read range of the Communicator. Numbers between **1** and **4** are possible, where 1 defines the shortest read range and 4 sets the longest read range.

B(number,1) The letter “B” followed by exactly 1 digit sets the baud rate and mode of serial communications. The possible numbers are:

0 for 4800 baud, RS232
1 for 9600 baud, RS232
2 for 19200 baud, RS232

L(xn,2) The letter *L* followed by the letter *F* or *V*.

F indicates fixed format. This means that the number of characters sent for barcode emulation port is fixed to *n*. Tag data that is shorter (i.e. has fewer digits) is preceded by 0_{ASCII} characters. Tag data that is longer (i.e. has more digits) is truncate after *n* digits.

V indicates variable format. This means that the number of characters sent for barcode emulation port depends on *n*. If the tag data is shorter (i.e., has fewer digits) only the necessary characters are sent. Tag data that is longer (more digits) is truncated after *n* digits. *n* is the length specification between 1 and 5.

Example:

Format	Number on Tag	Number on RS232 port-b
LF4	12	0012
	945	0945
	11018	1018
LV4	12	12
	945	945
	11018	1018
LF2	12	12
	945	45
	11018	18
LV2	12	12
	945	45
	11018	18

T (ASCII,2) The letter *T* followed by two ASCII characters specifying the hex value for the postamble

Example: T0D specifies a <CR> trailer as postamble
 T00 specifies that no postamble is sent

3.2) Reading and Reporting of Tags

After power up, the Custom Application will read all tags as fast as possible. If a tag enters the field it is read immediately. After a 500 ms “dead time” the tag is read again.

After a tag has been read and verified (only tags formatted to **Mini-Memory** are allowed) it is immediately reported via the serial interface. If the tag remains in the read zone it is not reported again. If a trigger occurs while the tag is in the read zone, emulated barcode data with possible postamble characters will be sent to the serial port-b interface.

If a tag leaves the read zone for more than 1s it is considered a new tag and reported again via serial interface. A new tag is always reported immediately, even if it enters the communications zone a few milliseconds after a previously read tag. In the data handling, 8 bits are sent in the first byte and the remaining 6 bits are sent in the second Byte. Bit 7 of the second byte is enabled for low battery and bit 8 is used for the “g” command.

In *enhanced mode* the 16-bit data string is sent to the RS232 port-a and emulated barcode data will be sent to the serial port-b if a trigger occurs while the tag is in the read zone. Please note a tag that enters the read zone of the communicator while the input is high is not sent to serial port-b.

a.) Digital Outputs

The digital outputs on the Communicator will be used to indicate that tag data is available on the serial interface. These digital outputs will remain high for 100 ms after the data word was sent out. In addition to those digital outputs, two so-called LED-outputs are used as an *alive* or *health* indicator. After power up the two LED outputs “blink” once/second. If desired the two *alive* signals can be read by a PLC.

b.) Digital Input

Digital input 1 on the Communicator is used to request the last data from the Communicator. Once the Custom Application receives a *high* on this input the most recent data word is sent to the serial interface again.

Digital input 2 on the Communicator is used to request a re-read. If a tag is in the field when the digital input 2 is selected, the data on that tag will be reported. Instead the result of this operation reflects the current state of the system. If no tag is in the field the *No Tag* reply (bf_{hex}) will be issued after 1 second.

c.) g<cr> - Command

Data can also be requested via the g-command. When this command is issued before the Communicator reads any tag it will respond with bf_{hex} .

To issue this command the Communicator must receive g<cr> equal to $67_{hex}0D_{hex}$. This command can be issued via the serial ports a or b.

The function of this command is identical to setting input 1, always old data, Bit 15=1.

d.) n<cr> - Command

Data can also be requested via the n-command. The function of this command is identical to setting input 2. The new data bit is always low.

To issue this command the Communicator must receive n<cr> equal to $6E_{hex}0D_{hex}$. This command can be issued via the serial port-a, and port-b.

e.) s<cr> - Command

Data can also be requested via the s-command. The function of this command is that it will return the string of data sent on boot up.

To issue this command the communicator must receive s<cr> equal to 73_{hex}0D_{hex}. This command can be issued via the serial port-a, and port-b.

f.) f<cr> - Command

Data can also be requested via the f-command. The function of this command is that it returns the same information as the g<cr> except the two-character tag format is added to the end. The tag format can be one of the following: **C0, C4, C8, C6, R4, R8, R6**

To issue this command the communicator must receive f<cr> equal to 66_{hex}0D_{hex}. This command can be issued via serial port-a, and port-b.

4) Track Mode Operation

The track mode operation is an extension of both the standard and enhanced modes. The purpose of track mode is to fully emulate an existing barcode based system. This enables the P+F long range ID system to be used as a drop-in replacement.

The following push button configuration and parameter settings are available in track mode below:

Display Characters	Parameter Value	Scope	Description
OP	3		Track mode
cF	01 to 99	All modes *(Default value of microwave channel = 50)	Microwave Channel
cr	1 to 4	All modes *(Default value of range setting= 1)	Range setting
br	0	All Modes	4800 baud
	1		9600 baud
	2		19200 baud
Pb	23	All Modes	RS232
	48	Standard, track, and universal mode	RS485
	42	Standard, track, and universal mode	RS422
FF	F	Enhanced, track and universal mode	Fixed length reply
	U		Variable length reply
Fn	1 to 5	Enhanced, track and universal mode	Length of reply
Ft	00 to FF	Enhanced, track and universal mode *(Default value of footer= <CR>)	Hex value of terminator
Fh	00 to FF	Track and universal mode *(Default value of header= 21 Hex)	Hex value of header
FS	00 to FF	Track mode *(Default value of handshake= NONE)	Hex value of handshake
rn	0	Track mode	<i>Read Always</i> , LON and LOFF are ignored
	1 to 9		Number of reads
Sc	NO	All Modes	DO NOT write configuration
	DO		Write configuration to tag

4.1) Format of the Set-Up Tag for Track Mode

On power up, the Custom Application is looking for tags in the read zone. Once a tag has been read, the data from that string is evaluated and tested for validity. A valid set-up tag must be formatted as **Full-Memory**. This procedure assures that only true set up tags are used. All tags that do not contain data compatible with the format of the set-up tags are rejected. A valid set-up tag must have the following format: ****(Set up tags only read on power-up)**

**\$\$\$\$S(text,5)C(number,2)R(number,1)B(number,1)L(xn,2)T(ASCII,2)H(ASCII,2)
X(ASCII,2)M(number,1)**

\$\$\$\$ This is a user definable string that precedes the set-up data. Its sole purpose is to make the data string unique to that communicator.

S(text,5) The letter “S” followed by exactly 5 characters is used as a station ID and is stored in memory. Currently, the station ID is not used at other places in the Custom Application.

C(number,2) The letter “C” followed by exactly 2 digits defines the microwave channel used for communications. The range of possible microwave channels ranges from **01** to **99**. This, being one of the most powerful features of the Ident-M System T Communicators, allows them to be placed in close proximity to one another without the chance of interference. The selection of the microwave channel has no effect on their ability to read a tag.

R(number,1) The letter “R” followed by exactly 1 digit sets the read range of the Communicator. Numbers between **1** and **4** are possible, where 1 defines the shortest read range and 4 sets the longest read range.

B(number,1) The letter “B” followed by exactly 1 digit sets the baud rate for serial communications. The possible numbers are:

0 for 4800 baud, RS232	5 for 4800, RS485	8 for 4800, RS422
1 for 9600 baud, RS232	6 for 9600, RS485	9 for 9600, RS422
2 for 19200 baud, RS232	7 for 19200, RS485	A for 19200, RS422

L(xn,2) The letter *L* followed by the letter *F* or *V*.
F indicates fixed format. This means that the number of characters sent for barcode emulation port is fixed to *n*. Tag data that is shorter (i.e. has fewer digits) is preceded by 0_{ASCII} characters. Tag data that is longer (i.e. has more digits) is truncate after *n* digits.
V indicates variable format. This means that the number of characters sent for barcode emulation port depends on *n*. If the tag data is shorter (i.e., has fewer digits) only the necessary characters are sent. Tag data that is longer (more digits) is truncated after *n* digits. *n* is the length specification between 1 and 5.

Example:

Format	Number on Tag	Number on RS232 port-b
LF4	12	0012
	945	0945
	11018	1018
LV4	12	12
	945	945
	11018	1018
LF2	12	12
	945	45
	11018	18
LV2	12	12
	945	45
	11018	18

- T (ASCII,2) The letter *T* followed by two ASCII characters specifying the hex value for the postamble
Example: T0D specifies a <CR> trailer as postamble
 T00 specifies that no postamble is sent
- H(ASCII,2) The letter *H* followed by two ASCII characters specifying the hex value for the preamble
Example: H0A specifies a <LF> header as preamble
 H00 specifies that no preamble is sent
- X(ASCII,2) The letter *X* followed by two ASCII characters specifying the hex value for the handshake character. The handshake confirms that a command has been sent and received by the communicator. This character is sent after a command (“LON”, “LOFF”, “g”) was received.
Example: X21 specifies a ! as the hand shake character
 X00 specifies that no hand shaking is used
- M(number,1) The letter *M* followed by exactly 1 digit sets the number of different tags that are read before the reading stops. A value of zero indicates that the Communicator starts scanning at power-up. The LON and LOFF commands are not executed.
Example: M1 specifies that reading tags stops after one tags has been read
 M0 specifies that reading always takes place LON and LOFF are ignored.

4.2) **Commands and Reporting of Tags**

The Control PC issues the following command to instruct the Communicator to scan for a tag (via port-a)

LON<cr>

The Communicator requires only one complete LON<cr> sequence to accept the command. A confirmation is sent (or not sent) confirming that the command was received depending on the setting of the X-parameter (exchange confirmation). Once the LON<cr> command is received, the Communicator scans for tags until one tag has been read (M=1). If the M-Parameter is set to “n”, the communicator will scan until it sees “n” different tags (i.e. different data). Successive tags are reported at least 500ms apart. Alternatively, the Control PC can stop scanning by sending

LOFF<cr>

The Communicator requires only one complete LOFF<cr> sequence to accept the command. A confirmation is sent (or not sent) depending on the X-parameter.

When a tag is read, it is sent out the serial ports (port-a and port-b) immediately. The data is preceded by a preamble specified via the H-parameter (Header) and followed by a postamble specified by the T-parameter (Trailer). Setting the preamble to H00 causes the Communicator to not send a preamble. Also, setting the postamble to T00 causes the Communicator to not send a postamble.

a.) Digital Outputs

The digital outputs on the Communicator will be used to indicate that tag data is available on the serial interface. These digital outputs will remain high for 100 ms after the data word was sent out. In addition to those digital outputs, two so-called LED-outputs are used as an *alive* or *health* indicator. After power up the two LED outputs “blink” once/second. If desired the two *alive* signals can be read by a PLC.

b.) Digital Input

Digital input 1 on the Communicator is used to request the last data from the Communicator. Once the Custom Application receives a *high* on this input the most recent data word is sent to the serial interface again.

Digital input 2 on the Communicator is used to request a re-read. If a tag is in the field when the digital input 2 is selected, the data on that tag will be reported. Instead the result of this operation reflects the current state of the system. If no tag is in the field the *No Tag* reply (bf_{hex}) will be issued after 1 second.

c.) g<cr> - Command

Data can also be requested via the g-command. When this command is issued before the Communicator reads any tag it will respond with bf_{hex} .

To issue this command the Communicator must receive g<cr> equal to $67_{hex}0D_{hex}$. This command can be issued via the serial ports a or b.

The function of this command is identical to setting input 1, always old data, Bit 15=1.

d.) n<cr> - Command

Data can also be requested via the n-command. The function of this command is identical to setting input 2. The new data bit is always low.

To issue this command the Communicator must receive n<cr> equal to $6E_{hex}0D_{hex}$. This command can be issued via the serial port-a, and port-b.

e.) s<cr> - Command

Data can also be requested via the s-command. The function of this command is that it will return the string of data sent on boot up.

To issue this command the communicator must receive s<cr> equal to $73_{hex}0D_{hex}$. This command can be issued via the serial port-a, and port-b.

f.) f<cr> - Command

Data can also be requested via the f-command. The function of this command is that it returns the same information as the g<cr> except the two-character tag format is added to the end. The tag format can be one of the following: **C0, C4, C8, C6, R4, R8, R6**

To issue this command the communicator must receive f<cr> equal to $66_{hex}0D_{hex}$. This command can be issued via serial port-a, and port-b.

5) Universal Mode Operation

The universal mode operation adds additional control over the data received from the communicator and also enables writing data to tags.

In universal mode is possible to buffer a settable number of tags and request either the last read tag or all buffered tag data. The tag data buffer is FIFO.

A simple command can now be used to write data to tags. This command also performs the necessary formatting operation with user selectable frame intervals.

A heartbeat function has been implemented. If no other data exchange (via the serial interface) has occurred one minute the Communicator send an unsolicited heartbeat.

The following push button configuration and parameter settings are available in universal mode below:

Display Characters	Parameter Value	Scope	Description
OP	4		Universal mode
cF	01 to 99	All modes *(Default value of microwave channel = 50)	Microwave Channel
cr	1 to 4	All modes *(Default value of range setting= 1)	Range setting
br	0	All Modes	4800 baud
	1		9600 baud
	2		19200 baud
Pb	23	All Modes	RS232
	48	Standard, track, and universal mode	RS485
	42	Standard, track, and universal mode	RS422
FF	F	Enhanced, track and universal mode	Fixed length reply
	U		Variable length reply
Fn	1 to 5	Enhanced, track and universal mode	Length of reply
Ft	00 to FF	Enhanced, track and universal mode *(Default value of footer=<CR>)	Hex value of terminator
Fh	00 to FF	Track and universal mode *(Default value of header= 21 Hex)	Hex value of header
FS	00 to FF	Track mode *(Default value of handshake= NONE)	Hex value of handshake
hb	0	Universal mode	Heartbeat disabled
	1		Unsolicited heartbeat (1min)
	2		Heartbeat with Handshake
Sc	NO	All Modes	DO NOT write configuration
	DO		Write configuration to tag

5.1) Format of the Set-Up Tag for Universal Mode

On power up, the Custom Application is looking for tags in the read zone. Once a tag has been read, the data from that string is evaluated and tested for validity. A valid set-up tag must be formatted as **Full-Memory**. This procedure assures that only true set up tags are used. All tags that do not contain data compatible with the format of the set-up tags are rejected. A valid set-up tag must have the following format: ******(Set up tags only read on power-up)

**UUUUS(text,5)C(number,2)R(number,1)B(number,1)L(xn,2)T(ASCII,2)H(ASCII,2)
A(ASCII,1)**

- UUUU** This is a generic string that precedes the set-up data. Its sole purpose is to make the data string unique
- S(text,5)** The letter “S” followed by exactly 5 characters is used as a station ID and is stored in memory. Currently, the station ID is not used at other places in the Custom Application.
- C(number,2)** The letter “C” followed by exactly 2 digits defines the microwave channel used for communications. The range of possible microwave channels ranges from **01** to **99**. This, being one of the most powerful features of the Ident-M System T Communicators, allows them to be placed in close proximity to one another without the chance of interference. The selection of the microwave channel has no effect on their ability to read a tag.
- R(number,1)** The letter “R” followed by exactly 1 digit sets the read range of the Communicator. Numbers between **1** and **4** are possible, where 1 defines the shortest read range and 4 sets the longest read range.
- B(number,1)** The letter “B” followed by exactly 1 digit sets the baud rate for serial communications. The possible numbers are:

0 for 4800 baud, RS232	5 for 4800, RS485	8 for 4800, RS422
1 for 9600 baud, RS232	6 for 9600, RS485	9 for 9600, RS422
2 for 19200 baud, RS232	7 for 19200, RS485	A for 19200, RS422

- L(xn,2)** The letter *L* followed by the letter *F* or *V*.
F indicates fixed format. This means that the number of characters sent for barcode emulation port is fixed to *n*. Tag data that is shorter (i.e. has fewer digits) is preceded by 0_{ASCII} characters. Tag data that is longer (i.e. has more digits) is truncate after *n* digits.
V indicates variable format. This means that the number of characters sent for barcode emulation port depends on *n*. If the tag data is shorter (i.e., has fewer digits) only the necessary characters are sent. Tag data that is longer (more digits) is truncated after *n* digits. *n* is the length specification between 1 and 5.

Example:

Format	Number on Tag	Number on RS232 port-b
LF4	12	0012
	945	0945
	11018	1018
LV4	12	12
	945	945
	11018	1018
LF2	12	12
	945	45

	11018	18
LV2	12	12
	945	45
	11018	18

T (ASCII,2) The letter *T* followed by two ASCII characters specifying the hex value for the postamble
Example: T0D specifies a <CR> trailer as postamble
 T00 specifies that no postamble is sent

H(ASCII,2) The letter *H* followed by two ASCII characters specifying the hex value for the preamble
Example: H0A specifies a <LF> header as preamble
 H00 specifies that no preamble is sent

A(ASCII,1) The letter *A* followed by one ASCII characters specifying the type of Heartbeat
Example: A0 specifies the no Heartbeat will be used
 A1 specifies that an unsolicited Heartbeat (1 min) will be used
 A2 specifies that an unsolicited Heartbeat (1 min) with handshake will be used

5.2) Commands and Reporting of Tags

When a tag is read, it is sent out the serial ports (port-a and port-b) immediately. The data is preceded by a preamble specified via the H-parameter (Header) and followed by a postamble specified by the T-parameter (Trailer). Setting the preamble to H00 causes the Communicator to not send a preamble. Also, setting the postamble to T00 causes the Communicator to not send a postamble. A tag must leave the field of the communicator 10 sec before it be reported again (other modes 1 sec).

The Control PC issues the following command to instruct the Communicator to scan for a tag (via port-a)

LON<cr>

Alternatively, the Control PC can stop scanning by sending

LOFF<cr>

Note: You are still able to write tags when sending a LOFF<cr>

a.) Digital Outputs

The digital outputs on the Communicator will be used to indicate that tag data is available on the serial interface. These digital outputs will remain high for 100 ms after the data word is sent out. In addition to those digital outputs, two so-called LED-outputs are used as an *alive* or *health* indicator. Once the Custom Application is executing the *Read-Module* the two LED outputs “blink” once/second. If desired, the two *alive* signals can be read by a PLC.

b.) Digital Input

Digital input 1 on the Communicator is used to request the last data from the Communicator. Once the Custom Application receives a *high* on this input the most recent data word is sent to the serial interface again.

f.) f<cr> - Command

Data can also be requested via the f-command. The function of this command is that it returns the same information as the g<cr> except the two-character tag format is added to the end. The tag format can be one of the following: **C0, C4, C8, C6, R4, R8, R6**

To issue this command the communicator must receive f<cr> equal to 66_{hex}0D_{hex}. This command can be issued via serial port-a, and port-b.

e.) a<cr> - Command

Data can also be requested via the a-command. The function of this command is identical to setting input 3. The new data bit is not set.

To issue this command the Communicator must receive a<cr> equal to 61_{hex}.0D_{hex}. This command can be issued via the RS232 port-a or RS485/422. Please note that RS485 is half-duplex and data collisions may occur.

f.) w(l)(data)(format)<cr> - Command

This command is used to write data to the tag. The allowed range for (data) is between 0 and 16383. For data protection reasons the length of (data) is specified in the (l) parameter, which can be between 1 and 5.

It is not necessary to add leading zeros. (format) specifies the rate at which the tags transmit the data when interrogated by the Communicator. The following possibilities can be used:

- C0 Continuous mode without frame interval. This results in the fastest possible read performance and is the suggested mode for all application where only one tag is in the read zone at any given point in time.
- C4 Continuous mode with 4 frame-times interval. In this mode it is possible to read 2 tags simultaneously.
- C8 Continuous mode with 8 frame-times interval. In this mode it is possible to read 4 tags simultaneously.
- C6 Continuous mode with 16 frame-times interval. In this mode it is possible to read 8 tags simultaneously.

- R4 Random mode with 4 frame-times interval. In this mode it is possible to read 4 tags simultaneously. This is the suggested format in all cases where a small number of tags could be in the read zone simultaneously. (Default)
- R8 Random mode with 8 frame-times interval. In this mode it is possible to read 8 tags simultaneously
- R6 Random mode with 16 frame-times interval. In this mode it is possible to read 16 tags simultaneously. This mode will result in very slow read performance and is only suggested in manual applications (i.e. door access operations.)

The Communicator will reply with a Status message:

- 0 Write OK
- 1 Write OK but Battery Low detected
- 4 Format Error
- 5 No Tag, Write failed

Example:

Writing a number to the tag that will be interpreted as 2245 when read as a barcode.

The 4 following the W indicates that (data) has a length of 4 bytes.

The tag is formatted to C0 for highest possible read performance

```
W42245C0<cr>
```

Writing a number to the tag that will be interpreted as 67 when read as a barcode. The tag is formatted to R4 to allow for “multiple-tags-infield.” The length of the data is 2 and is indicated by the 2 following the W.

```
W267R4<cr>
```

Heartbeat reply

To determine the health of the serial interface between Communicator and control PC the Communicator sends an unsolicited heartbeat if no other serial data has been sent to the PC for one minute. The format of the heartbeat follows the data format specification. The ‘H’ character is used to indicate the heartbeat. The length of the heartbeat is identical to the configured length of the data. In fixed length mode the exact number of characters including head and trailer will be sent. In variable mode only one byte will be sent.

Heartbeat mode 0 disables the heartbeat

Heartbeat mode 1 enables the 1-minute heartbeat

Heartbeat mode 2 enables the 1-minute heartbeat but also expects a user acknowledge.

In this mode the control PC must reply by sending a single H<cr> back to the Communicator. If this string is not received within 1 second, the Communicator indicates this by toggling the main LED between RED and AMBER (one second each.) The communicator continues to operate normally.

Toggling the Main LED is stopped as soon as the control PC sends the heartbeat data handshake reply.

Example:

Assuming the Communicator has been formatted to send a fixed number of 5 characters of barcode emulation data the heartbeat will be

```
<Header Byte>0HHHHH<Trailer Byte>
```

If Heartbeat mode =2 then the user must return H<cr> to the communicator

Assuming a variable format (with three bytes for instance) the heartbeat will be

```
<Header Byte>0H<Trailer Byte>
```

In both case the 0 indicates “everything OK” status.

The heartbeat handshake “H<CR>” may be sent anytime to reset the heartbeat timer.

6.) **Download Module**

The *download module* enables a new Custom Application to be downloaded into the Communicator. To speed up the download process the download baud rate is fixed at 19200 baud. For downloading to the communicator follow the directions below:

1. Close any other running applications except for the download program called (ConfiLoad v1.2 BETA 970411).
2. Connect PC and communicator using an RS232 serial cable. Ensure that communicator port is RS232a.
3. Power Cycle the communicator and when SW appears in the two segment LED display press the S2 push button one time. Now dL should appear in the two segment LED display.
4. In the download program enter file name *Mini_3_2.a21 (firmware downloading into communicator)*, then press enter.
5. Next enter the **COMM port** (COM1 or COM2) that you are using to communicate with the communicator, then press enter.
6. Enter the **Baud rate** 19200, then press enter.
7. Enter **Parity N**, then press enter.
8. Now press enter and the download will begin.
9. The download should take approx. 3-5 minute and it will display download finished at the end of the sequence. (**Warning:** *Do not interrupt download sequence or communicator will lock up.*)

6.1) **Set Up Tag**

How to write a set up tag:

******(The set up tag must be formatted to **Full-Memory** or the custom application will not write data to the tag.)

1. Toggle through the operation parameters using the S2 push button until Sc is displayed in the two segment LED display.
2. Now press the S3 push button 2 times, and d0 should be displayed in the two segment LED display.
3. Hold the data carrier over the communicator and press the S2 push button and you should hear a beep to let you know the set up tag has been written.

-----End of Document-----