SYSTEM MT

MTT6000-F180-B12-V45-MON
MTT3000-F120-B12-V45-MON

CE

Introduction

MTT…-…-B12-V45-MON comes with P+F specific monitor software. This software has been designed to fit many customers specific needs. It can be configured for 4 different modes of operation. They are standard, enhanced, track and universal mode. Read through the specific features and benefits of each mode to determine how you want to configure your reader.

Operating modes

Standard mode – This basic mode of operation is used to read and send data back to the various connected ports.

- **2 bytes, 19 bytes, or 71 bytes of binary coded data**
- **MTO read only tags supported**
- **No formatting**
- **Hardware or software triggers can be used to resend data or reread tags**

Enhanced mode – This extension of the basic mode allows for the formatting of tag data and a termination character to be specified. Formatted data is only sent on hardware trigger of input 1 to port 2. Port 2 data only sent when tag is over reader and a trigger occurs on Input 1.

- **2 bytes, 19 bytes, or 71 bytes of binary coded data sent out port 1**
- **MTO read only tags supported**
- **Fixed or variable length data sent out port 2**
- **Specification of termination character on port 2**
- **Data on port 2 only sent when tag is over reader and there is a trigger on input 1.**

Track mode – This mode of operation uses features of the enhanced mode but with additional parameters to further emulate a barcode reader.

- **The same formatted data is sent out both ports**
- **MTO read only tags supported**
- **Data length can be specified**
- **Specifying of termination character is possible**
- **Specifying of prefix character is possible**
- **LON and LOFF barcode read commands implemented, Optional. LON turns reader on for reading of 1 up to 9 different tags. LOFF can be sent any time to turn off reading.(M1 to M9 parameter only)**
- **Optional acknowledge character can be specified in response to LON and LOFF commands**

Universal mode – Some additional control over the serial data is now possible. This includes a status byte preceding the data, a heartbeat, and the storing of buffered data.

- **The same formatted data is sent out both ports**
- **MTO read only tags supported**
- **Data length can be specified**

- **Specifying of termination character possible**
- **Specifying of prefix character possible**
- **LON and LOFF barcode read commands implemented, Optional. Turns on and off reading of tags.**
- **Status byte proceeds tag data string**
- **Request Buffer command. Will return the last 10 tags read.**
- **Heartbeat can be sent at regular intervals with/without acknowledge**

Tag data format

All tags, accept read only tags, are formatted during the write data operation. The tag is written completely every time. It is not possible to write only part of the tag while leaving some data unchanged. If some data must not change read the entire tag first change the data and then write back entire tag. There are three memory models for the MTM type tags. The data can be 2 bytes long, 19 bytes long or 71 bytes long. Depending on the reader mode setting the length of the data can be truncated to smaller more manageable fixed length sizes.

Quarter and Full memory tags
Any length of 19 bytes or less will be formatted to Quarter memory and 20 bytes and higher is FULL memory. The smaller the memory model the faster the data can be sent to the reader. So Mini is faster then Quarter which is faster the FULL. If a length is specified for quarter and full memory that is smaller then the memory possible then the remaining bytes are filled with NULL 0x00.

Memory length options

Memory model	Length	Data range	Data type	Write length
Mini memory	2	0 to 16383	ASCII	1 to 5
Quarter memory	19	19 bytes	Binary/ASCII	001 - 019
Full memory	71	71 bytes	Binary/ASCII	001 - 071
MTO read only	9	:00000000 to 999999999	ASCII	-

During the write operation the tag format is also specified. The interval type is set to R for random and C for constant. This means that the time between successive data sent from the tag is at a constant time interval or a random time interval. Random should always be used if multiple tags in field are expected. C should be used for quickest response times. The interval length is then specified at 0, 4, 8, or 16. This is the time interval is 4, 8, or 16 times as long as the actual time it takes to send the data. The longer the interval the more likely you will be able to read more tags in field and also the longer the battery life.

Format options

Tag format	Interval type	Interval length	~multiple tags in field
C0	C	0	0
C4	C	4	0
C8	C	8	0
C16	C	16	0
R4	R	0 to 8	~4
R8	R	0 to 16	~8
R16	R	0 to 32	~8

Read/Write speed and battery life

Tag size	Format	Read time	Write time 99.99%	Battery life
Mini	C0	50ms	600ms	6
Mini	C4	100ms	800ms	6
Mini	C8	170ms	2100ms	10
Mini	C16	300ms	-	10
Quarter	C0	70ms	800ms	6
Quarter	C4	180ms	2300ms	9
Quarter	C8	350ms	4800ms	10
Quarter	C16	540ms	-	10
Full	C0	140ms	3000ms	6
Full	C4	370ms	6000ms	9
Full	C8	750ms	-	10
Full	C16	1110ms	-	10
MTO tags	Random	150ms max, 80ms average,	-	10

Writing data

When writing data to a tag there must be only one tag within the entire maximum read zone. If there are multiple tags in the field a write will not be performed. The write range is very short. The tag must be placed at a maximum of .25 m (.8ft) from the reader. Do not ever take the tag out of the write range before the write is complete. The tag will lockup and a special procedure will be necessary to unlock the tag. The writing of tags can take a long time. If no tag is in the read zone the write will timeout at about 13 seconds.

Writing MINI Memory tags.

Up to 2 bytes of data are formatted on the tag. The data is programmed in ASCII and can have a value from 0 up to 16383. Any write attempt outside this range will result in a status 4(bad command) returned. The length and tag format must also be specified. The length does not include the tag format or termination character.

Writing MINI tags	Command	w<L1><Data_Ascii_DEC><Format><CR>
	Response	<Status><CR>

Example	Command	w512345C8<CR>	
	Response	0<CR>	success
	Response	1<CR>	success, low battery
	Response	4<CR>	bad command, check string format
	Response	5<CR>	no tag in field

Writing QUARTER/FULL Memory tags.

A three-character length is used to specify the write data length. Unlike the mini memory case there is no restriction on the type of data that is written. The length does not include the tag format or termination character.

Writing Quarter/Full tags	Command	w<L3><Data_Binary><Format><CR>

	Response	<Status><CR>

Example	Command		w019abcdefghijklmnopqrsC8<CR>
	Response	0<CR>	success
	Response	1<CR>	success, low battery
	Response	4<CR>	bad command, check string format
	Response	5<CR>	no tag in field (Will take 13 seconds to return)

Ports

In this manual the ports are specified as port 1 or port 2 because on the reader and the data format may be different for each. The ports are defined as follows:

Port 1 RS232 port connected at J42-1 and Ethernet TCP/IP port 10000
Port 2 RS485 port connected at J41.1 and Ethernet TCP/IP port 10001
Ethernet Port 10006, An 8 digit number is sent out of the port every 2 seconds. It has a range from 00000002 to 99999999. On first boot up it starts at 00000002 and increments by 2 every 2 seconds. If the unit never shuts off it will take over 3.17 years to roll over.

Sending Commands

Commands used to configure and use this reader are simple. A serial timeout is used to determine which commands are valid. The entire command must arrive within 200 ms for the command to be valid. If not an error code 4<CR> will be sent. Single character commands are easy enough to send from the keyboard but all other commands should be sent from a text file or custom application.

Boot-up string

On power up the communicator will send the model number and version information as well as the current loaded configuration. If for some reason the data looks corrupt then you are probably connected at the wrong baud rate. Check all possible baud rates of 4800, 9600, 19200, 38400, 57600, and 115200 on your terminal program power cycling after each change. The default is 9600 bps. The communicator will take about 30s to boot-up. Here is an example of the boot-up data:

(C)P+F IDENT-M
MTT6000-F120-B12-V45-MON
#911894
1180055
23.09.08
SdebugC01R4B1LF5T23H26A2

Inputs and Outputs

The inputs are used to trigger a specific response from the reader. These responses are documents in the different modes Standard, Enhanced, Track, and Universal. The outputs are used to indicate to the user that data is being sent to the serial ports in a specific way.
Input 1:Used to send the last read tag to the serial port or the tag that is over the reader in Enhanced mode.
Input 2: Used to perform a reread of the tag data. Because tag data is sent only when an actual tag is over the reader the output is sent when a tag is read and sent to the serial port.
Input 3: Only used in Universal mode. The reader will respond with a list of all tags previously read up to 10. Once read the buffer is cleared. The output does not turn on when this data is sent to both serial ports.
Output 1: Turns on for 100 ms when new tag data is sent

Output 2: Turns on for 100 ms when new tag data is sent, same as output 1
Relay output: Turns on when a tag with low battery is read, independent of other outputs

Setting the IP address

If you plan on using the Ethernet ports the IP address and subnet mask will need to be set. The units come with a default IP address of 192.168.0.2 and a subnet mask of 255.255.255.0. The IP address can be set by logging into the web page of the unit. DHCP can be enabled here as well. Any of the serial ports can be used to read back the current IP address settings. This enables the user to reconnect to the unit if the IP address is forgotten.

Reading IP address

	Command	IP<CR>
	Response	ipaddress;subnetmask;gateway<CR>
Example	Command	IP<CR>
	Response	172.18.30.20; 255.255.0.0; 172.18.0.55<CR>

Web Interface

Logon to the web page using the following default user name and password:

Username: admin
Password: qwerty

Using the web interface the IP address, subnet mask and gateway can be set.

Configuration of the reader

The reader uses a setup string to configure itself for the correct mode of operation and parameter settings. This string is stored on the reader in a file. Typically the configuration of each reader is different. The microwave channel and the read range are the two most commonly configured parameters. Also each reader may have a unique name. The configuration can be modified in a number of ways including through serial/Ethernet ports, or setup tag. Here are examples of each.

1. Connect to the reader to ports 1 or 2 serially or via. Ethernet. Send the L command to the reader to read out the existing configuration. The response will start with a status “0” and a 3-digit length and end with <CR>. The C command can then be used to configure the readers. The response will be immediate and then the reader reboots automatically. The reader will go through a reboot and the normal boot up string can be used to verify the configuration. Capital letters must be used for all parameter data.

Read configuration

	Command	L<CR>
	Response	<Status><L3><ASCII_CAP_Data><CR>
Example	Command	L<CR>
	Response	0028UUUUStim05C01R4B1LF5T23H26A0<CR>

Write Configuration

	Command	C<L3><Configuration string><CR>
	Response	<Status><CR>
Example	Command	C028 UUUUStim05C01R4B1LF5T23H26A0<CR>
	Response	0<CR> + the reader will reboot automatically

2. Once the reader is configured the configuration can be written to a setup tag. If this tag is ever placed in front of the reader during the 30s boot up time the reader will be configured automatically. Special commands are used to write a tag with the current configuration of reader. There is also a command to read the setup tag. Once the tag data has the setup tag format it will not be reported as a data tag.

   Setup tags are very handy if you have to replace a reader. A separate setup tag should be used for each reader because every reader typically has a unique configuration. The new reader is wired up and the setup tag is placed in front of the reader on boot up. Configuration of the reader is completed and the reader runs normally.

Write setup tags

	Command	K<CR>
	Response	<Status><CR>
Example	Command	K<CR>
	Response	0<CR> if the tag is over the reader and write is successful

Read setup tags

	Command	M<CR>
	Response	<Status><L3><Configuration string><CR>
Example	Command	M<CR>
	Response	0028UUUUStim05C01R4B1LF5T23H26A0<CR>

Configuration string options

The configuration string holds all parameters to run the reader. Each mode of operation has a specific parameter set that must be included in the string. All parameters required for that operating mode must be in the string for the configuration to be valid otherwise an error code 4<CR> is returned.

$$$$ These are the first 4 characters of configuration string for standard, Enhanced and track modes.
UUUU These are the first 4 characters of configuration string for Universal mode

The default configuration is Universal mode with the following configuration string
SdebugC01R4B1LF5T23H26A2

	Standard	Enhanced	Track	Universal	Options	Default
S(text, 5)	*	*	*	*	Any 5 character text	Sdebug
C(num, 2)	*	*	*	*	Frequency channel 5-97 01,02 - frequency hopping default sub-bands 03,04 - adaptive frequency hopping default sub-bands 98 - frequency hopping all bands 99 - adaptive frequency hopping all bands	C01

					A1-P1 specific bands, frequency hopping A2-P2 specific bands, adaptive frequency hopping	
R(num,1)	*	*	*	*	Range 1-4 and A-J, 1 to 100% range	R4
B(num,1)	*	*	*	*	B0-BE 4800 bps to 115,200 bps, 2-wire or 4-wire RS485 Default – 9600 bps, 4-wire RS485	B1
L(xn,2)		*	*	*	F1 to F5 = fixed length, exactly 1 up to 5 characters V1 to V5 = variable length max 1 up to 5 characters 001 to 071 fixed length, exactly 1 up to 71 char F8 = :Mark only, FA is data :Mark	LF5
T(hex,2)		*	*	*	The hex value of the termination character default #	T23
H(hex,2)			*	*	The hex value of the prefix character default &	H26
X(hex,2)			*		The hex value of the handshake character	-
M(num,1)			*		1=one tag is scanned and reading stops 2=two tags are read and reading stops … 0=Reading always takes place	-
A(num,1)				*	0=no heartbeat 1=unsolicited heartbeat every 1 min 2= unsolicited heartbeat every 1 min and handshake required	A2

Examples of configuration of different modes

Standard mode station “str05”, microwave channel 30, baud rate 9600, Range 2
$$$$Sstr05C30R2B1

Enhanced mode station “MMM06”, adaptive frequency hopping default bands, baud rate 9600, range 4, Fixed format length 10, terminator carriage return
$$$$SMMM06C03R4B1L10T0D

Track mode station “12345”, channel 10, baud rate 9600, range 1, Fixed length 5, no terminator, no prefix, no handshake, one tag read after LON issued
$$$$S12345C10R1B1LF5T00H00X00M1

Universal mode station “abcde”, channel 97, baud rate 9600, Fixed length 71. Terminator carriage return, no prefix, with heartbeat and no user acknowledge
UUUUSabcdeC97R4B1L71T0DH00A1

Detail explanation of configuration options

S(text,5) The letter “S” followed by exactly 5 characters is used as a station ID and is stored in memory.

C(number,2) The letter “C” followed by exactly 2 digits defines the microwave channel used for communications. This is a powerful feature that allows many readers to work in close proximity to one another without interference.

Individual readers must be set to different frequencies if they are being used in the same plant in close proximity to one another. It may be beneficial to set readers with a channel gap of 2 or more to minimize mutual interference. Certain frequencies may be used by other hardware in the plant and these frequency

bands must be avoided or read/write range may be very low. An alternative would be to use Frequency hopping or adaptive frequency hopping. Frequency hopping will allow the reader to change its frequency automatically when reading. If there is interference and a read is unsuccessful then the next time the frequency changes the read may be successful. Adaptive frequency hopping will actually allow the reader to listen at a specific frequency before using it. If that frequency is taken then it will try another.

These readers cannot write while frequency hopping. If frequency hopping is being used it is turned off and the last channel it was on before it hopped will be used for the write functionality. If one unit is always used for writing then a fixed frequency channel should be used.

The fixed Channel settings 5-97 has a frequency separation of 300 kHz. The individual channels used when frequency hopping are separated by 200 kHz.

Fixed Frequency Channels

Channel	Frequency GHz	Channel	Frequency GHz	Channel	Frequency GHz	Channel	Frequency GHz
05	2.4362	30	2.4437	55	2.4512	80	2.4587
06	2.4365	31	2.4440	56	2.4515	81	2.4590
07	2.4368	32	2.4443	57	2.4518	82	2.4593
08	2.4371	33	2.4446	58	2.4521	83	2.4596
09	2.4374	34	2.4449	59	2.4524	84	2.4599
10	2.4377	35	2.4452	60	2.4527	85	2.4602
11	2.4380	36	2.4455	61	2.4530	86	2.4605
12	2.4383	37	2.4458	62	2.4533	87	2.4608
13	2.4386	38	2.4461	63	2.4536	88	2.4611
14	2.4389	39	2.4464	64	2.4539	89	2.4614
15	2.4392	40	2.4467	65	2.4542	90	2.4617
16	2.4395	41	2.4470	66	2.4545	91	2.4620
17	2.4398	42	2.4473	67	2.4548	92	2.4623
18	2.4401	43	2.4476	68	2.4551	93	2.4626
19	2.4404	44	2.4479	69	2.4554	94	2.4629
20	2.4407	45	2.4482	70	2.4557	95	2.4632
21	2.4410	46	2.4485	71	2.4560	96	2.4635
22	2.4413	47	2.4488	72	2.4563	97	2.4638
23	2.4416	48	2.4491	73	2.4566		
24	2.4419	49	2.4494	74	2.4569		
25	2.4422	50	2.4497	75	2.4572		
26	2.4425	51	2.4500	76	2.4575		
27	2.4428	52	2.4503	77	2.4578		
28	2.4431	53	2.4506	78	2.4581		
29	2.4434	54	2.4509	79	2.4584		

Frequency Hopping Band Settings

Channel Setting	Bands	Hopping
A1	Sub-band “A”: 2402.0–2406.8 MHz	Frequency Hopping
B1	Sub-band “B”: 2407.0–2411.8 MHz	Frequency Hopping

C1	Sub-band “C”: 2412.0–2416.8 MHz	Frequency Hopping
D1	Sub-band “D”: 2417.0–2421.8 MHz	Frequency Hopping
E1	Sub-band “E”: 2422.0–2426.8 MHz	Frequency Hopping
F1	Sub-band “F”: 2427.0–2431.8 MHz	Frequency Hopping
G1	Sub-band “G”: 2432.0–2436.8 MHz	Frequency Hopping
H1	Sub-band “H”: 2437.0–2441.8 MHz	Frequency Hopping
I1	Sub-band “I”: 2442.0–2446.8 MHz	Frequency Hopping
J1	Sub-band “J”: 2447.0–2451.8 MHz	Frequency Hopping
K1	Sub-band “K”: 2452.0–2455.8 MHz	Frequency Hopping
L1	Sub-band “L”: 2457.0–2461.8 MHz	Frequency Hopping
M1	Sub-band “M”: 2462.0–2466.8 MHz	Frequency Hopping
N1	Sub-band “N”: 2467.0–2471.8 MHz	Frequency Hopping
O1	Sub-band “O”: 2472.0–2476.8 MHz	Frequency Hopping
P1	Sub-band “P”: 2477.0–2481.8 MHz	Frequency Hopping
A2	Sub-band “A”: 2402.0–2406.8 MHz	Adaptive Frequency Hopping
B2	Sub-band “B”: 2407.0–2411.8 MHz	Adaptive Frequency Hopping
C2	Sub-band “C”: 2412.0–2416.8 MHz	Adaptive Frequency Hopping
D2	Sub-band “D”: 2417.0–2421.8 MHz	Adaptive Frequency Hopping
E2	Sub-band “E”: 2422.0–2426.8 MHz	Adaptive Frequency Hopping
F2	Sub-band “F”: 2427.0–2431.8 MHz	Adaptive Frequency Hopping
G2	Sub-band “G”: 2432.0–2436.8 MHz	Adaptive Frequency Hopping
H2	Sub-band “H”: 2437.0–2441.8 MHz	Adaptive Frequency Hopping
I2	Sub-band “I”: 2442.0–2446.8 MHz	Adaptive Frequency Hopping
J2	Sub-band “J”: 2447.0–2451.8 MHz	Adaptive Frequency Hopping
K2	Sub-band “K”: 2452.0–2455.8 MHz	Adaptive Frequency Hopping
L2	Sub-band “L”: 2457.0–2461.8 MHz	Adaptive Frequency Hopping
M2	Sub-band “M”: 2462.0–2466.8 MHz	Adaptive Frequency Hopping
N2	Sub-band “N”: 2467.0–2471.8 MHz	Adaptive Frequency Hopping
O2	Sub-band “O”: 2472.0–2476.8 MHz	Adaptive Frequency Hopping
P2	Sub-band “P”: 2477.0–2481.8 MHz	Adaptive Frequency Hopping
01 or 02	Bands G,H,I,J,K,L	Frequency Hopping
03 or 04	Bands G,H,I,J,K,L	Adaptive Frequency Hopping
98	All bands	Frequency Hopping
99	All bands	Adaptive Frequency Hopping

R(number,1) The letter “R” followed by exactly 1 digit sets the read range of the Communicator. Numbers between **1**-**4** and A-J are possible, where 1 defines the shortest read range and 4 sets the longest read range. The read range can be set in 10% increments using the settings A(10%) up to J(100%). Range 4 and J have the same range.

Range Setting	% of Maximum distance
1	1%
2	33%
3	66%
4	100%
A	10%
B	20%

C	30%
D	40%
E	50%
F	60%
G	70%
H	80%
I	90%
J	100%

B(number,1) The letter "B" followed by exactly 1 digit sets the baud rate for both serial ports and the port 2 RS-485 mode. Because 2-wire mode is half-duplex data collisions are possible. The possible settings are:

Setting	Baud rate(bps)	Port 1 setting	Port 2 setting	Port 2 mode
B0 or B8	4800	RS232	RS485	4-wire
B1 or B9	9600	RS232	RS485	4-wire
B2 or BA	19200	RS232	RS485	4-wire
B3	38400	RS232	RS485	4-wire
B4	57600	RS232	RS485	4-wire
BB	115200	RS232	RS485	4-wire
B5	4800	RS232	RS485	2-wire
B6	9600	RS232	RS485	2-wire
B7	19200	RS232	RS485	2-wire
BC	38400	RS232	RS485	2-wire
BD	57600	RS232	RS485	2-wire
BE	115200	RS232	RS485	2-wire

L(xn,2) The length and data format can be set using the letter *L* followed by the letter *F* or *V* and length.

Fn 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. For Quarter and full memory tags n digits are always sent. The remaining bytes are truncated.

F8 is a special case where only the MARK is read off of read/write and read only tags. The data starts with : and is followed by an 8 digit MARK.

Ex. :12345678<CR>

FA is a special case where the entire tag data is sent and the MARK is sent after. The datasets are separated by a ":".

MARK only tags return 9 characters on port 1 and 14 characters on 2
MINI memory tags return 11 characters
QUARTER memory tags return 28 characters
FULL memory tags return 80 characters

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. For Quarter and full memory tags n digits are always sent. The remaining bytes are truncated.

xx length specifier can also be Lxx where xx is the fixed length of data to be sent.

The read only(MTO) tags always have a fixed format of : and 8 characters.

Example:

Format	Number on Tag	Data sent when triggered
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
L20	12(Mini)	00000000000000000012
	abcdefghijklmnopqrs(Quarter)	Abcdefghijklmnopqrs<0x00>
	xxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxx(Full)	xxxxxxxxxxxxxxxxxxx
MTO tags	:12018628(Read only, MTO)	:12018628
LF8	:12018628(Any tag)	:12018628
LFA	12(Mini)	00012:01474380
	abcdefghijklmnopqrs(Quarter)	Abcdefghijklmnopqrs: 01474380
	xxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxx:014743 80 (Full)	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxx:01474380

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.

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

Data format for each mode

Standard mode

The format of the data and the responses to the various commands are as follows.

Tag moves over reader, no trigger or command sent

The length of the data will depend on what format the tag is in. Two bytes are sent for MINI, 18 bytes for QUARTER and 71 bytes for FULL memory tags. Read only tag data, 9 bytes, is preceeded by a colon.

Command No command sent

Mini/Quarter/Full memory tag read

Response Port 1 and 2 = <Data_Binary_Variable>(Bit 6 set if low battery on MINI tags)

Read only MTO tag read

Response Port 1 and 2 = :<Data_ASCII_DEC>

Input 1 triggered or G<CR> command sent, Requests the last dataset sent

The responses are sent of the last tag that was over the reader. If no tag has been read since boot up then the <0xBF><0xFF> message is sent. The length of the data will depend on what format the tag is in. Two bytes are sent for MINI, 18 bytes for QUARTER and 71 bytes for FULL memory tags. A colon precedes read only tag data, 9 bytes.

Command G<CR> or rising edge input 1

Mini/Quarter/Full memory tag read

Response Port 1 and 2 = <Data_Binary_Variable> (Bit 6 low bat, bit 7 on for MINI)

Read only MTO tag read

Response Port 1 and 2 = :<Data_ASCII_DEC>

No tag read since boot up

Response Port 1 and 2 = <0xBF><0xFF>

Input 2 triggered or N<CR> command sent, reread tag

Data is only sent if the tag is over the reader. If there is no tag over the reader the <0xBF><0xFF> response is sent. The length of the data will depend on what format the tag is in. Two bytes are sent for MINI, 18 bytes for QUARTER and 71 bytes for FULL memory tags. Read only tag data, 9 bytes, is preceded by a colon.

Command N<CR> or rising edge input 2

Mini/Quarter/Full memory tags
Response Port 1 = <Data_Binary_Variable>(Bit 6 low bat, bit 7 on for MINI)
Port 2 = <Data_Binary_Variable> (Bit 6 low bat on MINI)

Read only MTO tag over reader
Response Port 1 and 2 = :<Data_ASCII_DEC>

No tag over reader
Response Port 1 and 2 = <0xBF><0xFF>

F<CR> command sent, Requests the last data set sent and includes tag format

The responses are sent of the last tag that was over the reader. The tag format is appended to the end of the string examples include C0, R4, R8… If no tag has been read since boot up then the <0xBF><0xFF>?? message is sent. The length of the data will depend on what format the tag is in. Two bytes are sent for MINI, 18 bytes for QUARTER and 71 bytes for FULL memory tags. Read only tag data, 9 bytes, is preceded by a colon.

Command F<CR>

Mini/Quarter/Full memory tags
Response Port 1 and 2 = <Data_Binary_Variable><Format> (Bit 6 low bat, bit 7 on for MINI)

Read only MTO tag
Response Port 1 and 2 = :<Data_ASCII_DEC>R8

No tag read since bootup
Response Port 1 and 2 = <0xBF><0xFF>??

Enhanced Mode

Enhanced mode will add formatting to the MINI memory data and will allow the data to have a fixed length out of port 2.

Data format of tags with no triggers.

Port 1 data is sent immediately when the tag arrives in field
Port 2 no data is sent

Command No command sent

Mini/Quarter/Full memory tags
Response Port 1 = <Data_Binary_Variable>(Bit 6 set if low battery on MINI tags)
Port 2 = -

Read only MTO tag or all tags when LF8 string format is specified
Response Port 1 = : <Data_ASCII_DEC>
Port 2 = -

Input 1 triggered or G<CR> command sent, Send las tag read to port 1 and only send to port 2 if tag is currently over reader.
Port 1 will send the last data set that was sent from reader
Port 2 will only send the tag data that is over the reader

Command G<CR> or rising edge input 1

Mini memory tags
Response Port 1 = <Data_Binary_ Variable><TERM> (Bit 6 low bat, bit 7 on for MINI)
Port 2 =<Data_ASCII_DEC><TERM>(Length Formatted by Lxx parameter)

Quarter/Full memory tag
Response Port 1 = <Data_Binary_Variable><TERM>
Port 2 = <Data_Binary_Fixed><TERM> (length specified by Lxx parameter)

Read only MTO tag or all tags when LF8 is specified
Response Port 1 and 2 =:<Data_ASCII_DEC><TERM>

No tag read since bootup
Response Port 1 = <0xBF><0xFF><TERM>
Port 2 = -

Input 2 triggered or N<CR> command, reread tag
Data is only sent if the tag is over the reader. If there is no tag over the reader the <0xBF><0xFF><TERM> response is sent. The length of the data will depend on what format the tag is in. Two bytes are sent for MINI, 18 bytes for QUARTER and 71 bytes for FULL memory tags. Read only tag data, 9 bytes, is preceeded by a colon.

Command N<CR> or rising edge input 2

Mini/quarter/full memory tag over reader
Response Port 1 = <Data_Binary_Variable>(Bit 6 set if low battery on MINI tags)
Port 2 = -

Read only MTO tag over reader or all tags when LF8 is specified
Response Port 1 = : <Data_ASCII_DEC
Port 2 = -

No Tag over reader
Response Port 1 = <0xBF><0xFF><TERM>(3)
Port 2 = -

F<CR> command sent,Requests the last dataset sent include tag format
The responses are sent of the last tag that was over the reader. The tag format is appended to the end of the string examples include C0, R4, R8… If no tag has been read since boot up then

the <0xBF><0xFF><TERM>?? message is sent. The length of the data will depend on what format the tag is in. Two bytes are sent for MINI, 18 bytes for QUARTER and 71 bytes for FULL memory tags. Read only tag data, 9 bytes, is preceded by a colon. Length is specified by Lxx parameter except when reading the Mark.

Command F<CR>

Mini Memory Tag
Response Port 1 = <Data_Binary_Variable><TERM><Format> (Bit 6 low bat, bit 7 on for MINI)
Port 2 = <Data_ASCII_DEC><TERM>
Quarter/Full memory tag
Response Port 1 = <Data_Binary_Variable><TERM><Format>
Port 2 = <Data_Binary_Fixed><TERM>

Read only MTO tag or all tags when LF8 is specified
Response Port 1 = :<Data_ASCII_DEC><TERM><Format>
Port 2 = :<Data_ASCII_DEC><TERM>

No tag read since bootup
Response Port 1 = <0xBF><0xFF><TERM>??
Port 2 = -

Track Mode

Track mode is designed to completely emulate a barcode reader. The preamble and termination characters are optional and the length is specified by the L parameter. The reader can be turned on using the LON and LOFF commands. If M=0 then data is sent immediately when the tag arrives. If M NOT 0 then LON must be sent before tags will be sent to the ports. The value of M is the number of different tags that can be read(up to 9) before the reader will no longer respond with tag data. LOF can be sent anytime to terminate reads. LON must be resent when the number of tags reaches the number M or the LOFF was sent.

Data format of tags with no trigger
Data is sent out of the serial ports automatically when LON has been sent, or M parameter is 0. Length is specified by Lxx parameter except when reading the Mark.

Command no command sent

MINI memory tags
Response Port 1and 2 = <HEAD><Data_ASCII_DEC><TERM>

Quarter and Full memory tag
Response Port 1 and 2 = <HEAD><Data_Binary_Fixed><TERM>
Read only MTO tag or all tags when LF8 is specified
Response Port 1 and 2 = <HEAD>:<Data_ASCII_DEC><TERM>

Input 1 triggered or G<CR> command sent, request last data sent
The last tag read is sent out of both serial ports. Length is specified by Lxx parameter except when reading the Mark.

Command G<CR> or rising edge input 1

MINI memory tags
Response Port 1and 2 = <HEAD><Data_ASCII_DEC><TERM>

Quarter and Full memory tag
Response Port 1 and 2 = <HEAD><Data_Binary_Fixed><TERM>

Read only MTO tag or all tags when LF8 is specified
Response Port 1 and 2 = <HEAD>:<Data_ASCII_DEC><TERM>

No tag read since bootup
Response Port 1 and 2 =<HEAD>00000000…<TERM>
Port 1 and 2 =<HEAD>:00000000<TERM>(With F8 specified)
Port 1 and 2 =<HEAD>00000:00000000<TERM>(With FA specified)

Input 2 triggered or N<CR> command, reread tag
Any tag that is over the reader is sent out of the ports. Doesn't matter if LON was sent or that the communicator is reading tags. Length is specified by Lxx parameter except when reading the Mark.

Command N<CR> or rising edge of input 2

MINI memory tags
Response Port 1and 2 = <HEAD><Data_ASCII_DEC><TERM>

Quarter and Full memory tag
Response Port 1 and 2 = <HEAD><Data_Binary_Fixed><TERM>

Read only MTO tag or all tags when LF8 is specified
Response Port 1 and 2 = <HEAD>:<Data_ASCII_DEC><TERM>

No tag over reader normal
Response Port 1 and 2 =<HEAD>00000000…<TERM>
Port 1 and 2 =<HEAD>:00000000<TERM>(With F8 specified)
Port 1 and 2 =<HEAD>00000:00000000<TERM>(With FA specified)

F command sent, request last data sent include tag type
The responses are sent of the last tag that was over the reader. The tag format is appended to the end of the string examples include C0, R4, R8… If no tag has been read since boot up then the then a fixed length of "0" are sent. Length is specified by Lxx parameter except when reading the Mark.

Command F<CR>

MINI memory tags
Response Port 1and 2 = <HEAD><Data_ASCII_DEC><TERM><Format>

Quarter and Full memory tag
Response Port 1 and 2 = <HEAD><Data_Binary_Fixed><TERM><Format>

Read only MTO tag or all tags when LF8 is specified
Response Port 1 and 2 = <HEAD>:<Data_ASCII_DEC><TERM><Format>

No tag read since boot up normal

Response Port 1 and 2 =<HEAD>00000000…<TERM>??
Port 1 and 2 =<HEAD>:00000000<TERM>??(With F8 specified)
Port 1 and 2 =<HEAD>00000:00000000<TERM>??(With FA specified)

Universal Mode

The universal mode operation adds additional control over the data received from the communicator.
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 heartbeat function has been implements and is optional. If no other data exchange (via the serial interface) has occurred for one minute the Communicator sends an a heartbeat string. This is used to verify that the reader is still powered up and communicating.

Data format of tags with no triggers.
Data is sent out of the serial ports automatically when LON has been sent, or M parameter is 0. Length is specified by Lxx parameter except when reading the Mark.

Command no command sent

MINI memory tags
Response Port 1and 2 = <HEAD><Status><Data_ASCII_DEC><TERM>

Quarter and Full memory tag
Response Port 1 and 2 = <HEAD><Status><Data_Binary_Fixed><TERM>
Read only MTO tag or all tags when LF8 is specified
Response Port 1 and 2 = <HEAD>:<Status><Data_ASCII_DEC><TERM>

Input 1 triggered or G<CR> command sent, request last data set
The last tag read is sent out of both serial ports. Length is specified by Lxx parameter except when reading the Mark.

Command G<CR> or rising edge of input 1

MINI memory tags
Response Port 1and 2 = <HEAD>H<Data_ASCII_DEC><TERM>

Quarter and Full memory tag
Response Port 1 and 2 = <HEAD>H<Data_Binary_Fixed><TERM>

Read only MTO tag or all tags when LF8 is specified
Response Port 1 and 2 = <HEAD>H:<Data_ASCII_DEC><TERM>

No tag read since boot up
Response Port 1 and 2 =<HEAD>500000000…<TERM>
Port 1 and 2 =<HEAD>5:00000000<TERM>(With F8 specified)
Port 1 and 2 =<HEAD>500000:00000000<TERM>(With FA specified)

Input 2 triggered or N<CR> command, reread tag
Any tag that is over the reader is sent out of the ports. Doesn't matter if LON was sent or that the communicator is reading tags. Length is specified by Lxx parameter except when reading the Mark.

Command N<CR> or rising edge of Input 2

MINI memory tags
Response Port 1and 2 = <HEAD><Status><Data_ASCII_DEC><TERM>

Quarter and Full memory tag
Response Port 1 and 2 = <HEAD><Status><Data_Binary_Fixed><TERM>

Read only MTO tag or all tags when LF8 is specified
Response Port 1 and 2 = <HEAD><Status>:<Data_ASCII_DEC><TERM>

No tag over reader
Response Port 1 and 2 =<HEAD>500000000…<TERM>
Port 1 and 2 =<HEAD>:500000000<TERM>(With F8 specified)
Port 1 and 2 =<HEAD>500000:00000000<TERM>(With FA specified)

F command sent, request last dataset include tag type
The responses are sent of the last tag that was over the reader. The tag format is appended to the end of the string examples include C0, R4, R8… If no tag has been read since boot up then the then a fixed length of "0" are sent. Length is specified by Lxx parameter except when reading the Mark.

Command F<CR>

MINI memory tags
Response Port 1and 2 = <HEAD>H<Data_ASCII_DEC><TERM><Format>

Quarter and Full memory tag
Response Port 1 and 2 = <HEAD>H<Data_Binary_Fixed><TERM><Format>

Read only MTO tag or all tags when LF8 is specified
Response Port 1 and 2 = <HEAD>H:<Data_ASCII_DEC><TERM><Format>

No tag read since boot up normal
Response Port 1 and 2 =<HEAD>50000000…<TERM>??
Port 1 and 2 =<HEAD>5:00000000<TERM>??(With F8 specified)
Port 1 and 2 =<HEAD>500000:00000000<TERM>??(With FA specified)

Input 3 triggered or A<CR> sent, Request all buffered data
All previously read tags, up to 10, are sent out of the serial ports with a 500ms gap between each one. After the last tag is sent an end of buffer message is sent. The length of each response is specified by Lxx parameter except when reading the Mark. Once the response has been sent the buffer is cleared.

Command A<CR> or rising edge of input 3

MINI memory tags

Response Port 1and 2 = <HEAD>H<Data_ASCII_DEC><TERM>

Quarter and Full memory tag

Response Port 1 and 2 = <HEAD>H<Data_Binary_Fixed><TERM>

Read only MTO tag or all tags when LF8 is specified

Response Port 1 and 2 = <HEAD>H:<Data_ASCII_DEC><TERM>

End of buffer message

Response Port 1 and 2 =<HEAD>5##…<TERM>
Port 1 and 2 =<HEAD>5#########<TERM>(With F8 specified)
Port 1 and 2 =<HEAD>5#<TERM>(With FA, Vx specified)

Heartbeat

To determine the health of the serial interface between Communicator and control PC the reader 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, F1-F5 the exact number of characters including head and trailer will be sent. In all other cases 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.

Heartbeat mode 1

Sent by reader automatically

Heatbeat <HEAD>0H…<TERM>

Response -

Heartbeat mode 2

Sent by reader automatically

Heatbeat <HEAD>0H…<TERM>

Response H<cr>

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

Legend

<Data_Binary_Variable> This binary coded data will have a variable length that depends on the format of the tag. Mini Memory Tags have a length of 2, Quarter memory tags have a length of 19 and Full memory tags have a length of 71.

<Data_Binary_Fixed> This binary coded data will have a fixed length based on the Lxx parameter. LFx, LVx, and Lxx will have the data formatted to exactly that many characters. If there are fewer characters on the tag the string is padded with 0x00<NULL> for quarter/full memory tags. MINI tags are padded with a 0x30,"0", If the tag data is longer then the fixed length specified the remaining data is truncated. LFA is a special variable length case where the tag data, which depends on tag format, is followed by the mark.

<Data_Binary> This binary coded data will be sent in bytes and can have a byte value anywhere from 0x00 to 0xFF. Any character can be sent

<Data_ASCII_DEC> This ASCII data will only contain the digits 1 through 9. Read only MTO tags will have 8 characters with a value from 00000000 to 99999999. Mini memory formatted data will have a value from 00000 to 16383. Depending on the Variable of fixed length format leading digits may be truncated.

<Format> This is the format of tag that was configured when the write procedure was performed

C0	Constant interval of 0
C4	Constant interval length between frames of 4
C8	Constant interval length between frames of 8
C6	Constant interval length between frames of 16
R4	Random interval length between frames between 0 and 8
R8	Random interval length between frames between 0 and 16
R6	Random interval length between frames between 0 and 32

<L1> This is a one character length used to write MINI memory tags. Its value can be between 1 and 5 and is the length of data to follow

<L3> This is the 3 character length and could have a value from 001 to 071

<CR> This special character is a carriage return. It is a single character that has a decimal value of 13 or hex value of 0x0D. There is no linefeed included.

<0xBF><0xFF> Two characters sent when no tag is over the reader or no tag read after boot-up. Check specific mode for details. These

0	*Success*
1	*Success, low battery, replace tag*
5	*No Tag*
4	*Bad Command*

<HEAD> This is the one character preamble character that is sent as the first character of the response. It is specified by the Hxx parameter. If H00 is used then no character is sent.

<TERM> This is the one character postamble character that is sent as the last character of the response. It is specified by the Txx parameter. If T00 is used then no character is sent.