MIC 10

WARRANTY STATEMENT

Titan Logix Corp. warrants all equipment of its own manufacture to be free of defects in material and workmanship for twelve (12) months from date of shipment. Titan Logix Corp.'s sole obligation hereunder shall be expressly limited to repair or exchange free of charge, F.O.B. Edmonton, Alberta, Canada, of such defective equipment (alternatively, Titan Logix Corp. will, at its option, refund the purchase price). Titan Logix Corp.'s obligation under this warranty is limited to the above and does not apply to exchange or repairs, which are required as a result of improper installation, misuse, maladjustment, abnormal operating conditions, or lack of routine maintenance. Nor does the warranty include the furnishing of service for maintenance or problems arising from the foregoing causes. No claims for labour, installation, removal, transportation, or other expenses will be recognized. Notwithstanding any stipulation of the purchaser to the contrary, all other obligations, representations, warranties and conditions, express or implied, statutory or otherwise, including any implied warranties or conditions of merchantability, quality or fitness are hereby excluded and, Titan Logix Corp. shall not be liable for any loss, cost or damages, of any kind whatsoever, whether consequential, indirect, special or otherwise, arising out of or in connection with the equipment or any defect therein, even if caused by the negligence of Titan Logix Corp., its employees or agents. The provisions hereof relating to the warranty and limitations hereon and limitation of liability shall continue to be enforceable between the parties notwithstanding termination of the within agreement for any reason including fundamental breach. Equipment not of Titan Logix Corp. manufacture will carry the vendor's or manufacturer's standard warranty.

PROPRIETARY INFORMATION

The Information disclosed herein contains proprietary rights of Titan Logix Corp. Neither this document nor the information disclosed herein shall be reproduced or transferred to other documents, or used or disclosed to others for manufacturing purposes, or for any other purpose except as specifically authorized in writing by Titan Logix Corp.

WARNINGS

THIS EQUIPMENT IS SUITABLE FOR USE IN GENERAL PURPOSE LOCATIONS ONLY.

Table of Contents

1200-0014

1 Product Description

1.1 About This Manual

This instruction manual provides information specific to the Titan Logix Corp. MIC 10 Modem. Other peripheral equipment should be supplied with its own instruction manual and that manual should be referred to for proper operation of the peripheral equipment.

It is essential that this manual be read and understood for proper installation and operation of your new MIC 10.

This manual includes:

INTRODUCTION:	Briefly describes the key features of the product.
INSTALLATION:	Detailed description of mounting and wiring of external equipment.
OPERATION:	Describes the operation and use of the MIC 10.
FPROGRAMMING:	Describes the procedures for programming MIC 10.
TROUBLESHOOTING:	Describes several quick problem solving techniques.
SPECIFICATIONS:	Describes the physical and operational characteristics.

1.2 About the MIC 10

Main Features

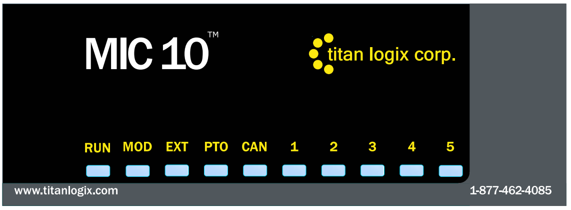
The MIC 10 is a protocol translator used to convert SV bus information from multiple TD80's to signal that can be sent to a 3rd party modem or a TD80 Single Display.

External Modem

Level, alarm, and error information collected by the MIC 10 will be forwarded to a modem via an RS232 port in programmable intervals. Two MIC 10 units can be chained together. Up to five TD80 Transmitters can be connected to each MIC 10 unit. This provides the capability to monitor up to ten TD80 Transmitters at one time.

TD80 Multi-Tank Display

The TD80 Multi-Tank Display receives data from the MIC 10 via a CAN port on the MIC 10 and the Display. The Display can receive and display level information for up to 5 TD80's connected to a MIC 10.



1.3 What's Included

A MIC 10 includes:

- A MIC 10 unit
- Mounting bracket
- RS232 cable
- Users manual

2 Installation

Trained Personnel

Installation should only be performed by qualified personnel and in accordance with local governing regulations.

Update Rate Jumper

Before mounting the MIC 10 the update rate should be set by changing the jumpers, if required, inside the MIC 10. The update rate sets the frequency that the MIC 10 will transmit data to the upstream modem and it is factory set to once every 5 minutes. These jumpers only need to be set on a MIC 10 connected directly to the modem. The jumpers can be accessed by removing the two recessed screws on the bottom of the MIC 10 an opening the cover.

When changing the jumper setting the following practices will minimize possible damage to the components on the board, resulting from electrostatic discharge or improper installation.

- ✓ Switch power off to unit before opening unit.
- ✓ Touch only the jumpers
- ✓ When removing the printed circuit boards handle only by non-conductive surfaces.
- ✓ Minimize the handling of printed circuit boards.

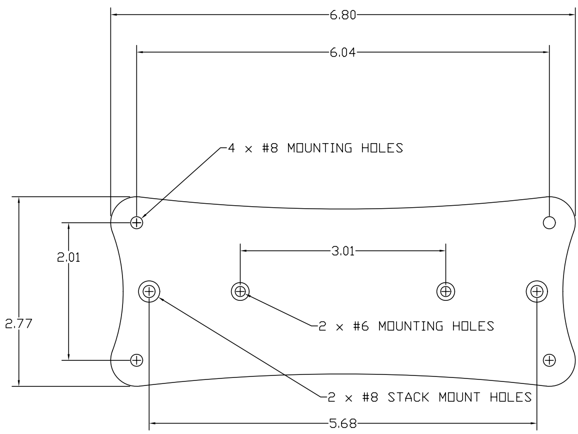
Below lists the jumper settings required for various update rates.

Jumper State				Update Rate
A	B	C	D	
Out	Out	Out	Out	1 second
In	Out	Out	Out	2 seconds
Out	In	Out	Out	10 seconds
In	In	Out	Out	1 minute
Out	Out	In	Out	2 minutes
In	Out	In	Out	3 minutes
Out	In	In	Out	4 minutes
In	In	In	Out	5 minutes
Out	Out	Out	In	6 minutes
In	Out	Out	In	7 minutes
Out	In	Out	In	8 minutes
In	In	Out	In	9 minutes
Out	Out	In	In	10 minutes
In	Out	In	In	15 minutes
Out	In	In	In	30 minutes
In	In	In	In	1 hour

2.1 Mounting

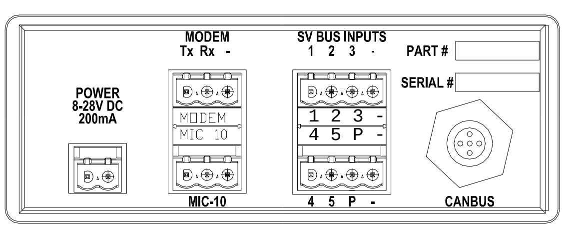
The MIC 10 needs be mounted away from the weather in a location were it will not be damaged. The select mounting location should allow easy access for wiring and to view the status indicating LED's on the front of the unit.

The MIC 10 comes with an aluminum mounting bracket that can be used to mount the MIC 10 in a variety of ways. The bracket can be mounted to the top or the bottom of the unit with 2 of the #6 screws provided. The unit can then be mounted as required using 4 - #8 mounting holes. Two or more units can be stacked up and attached together using the 2 - #8 stack mounting holes to attach units together.



2.2 Wiring

The following figure illustrates the terminal connections on the back of a MIC 10.



Power

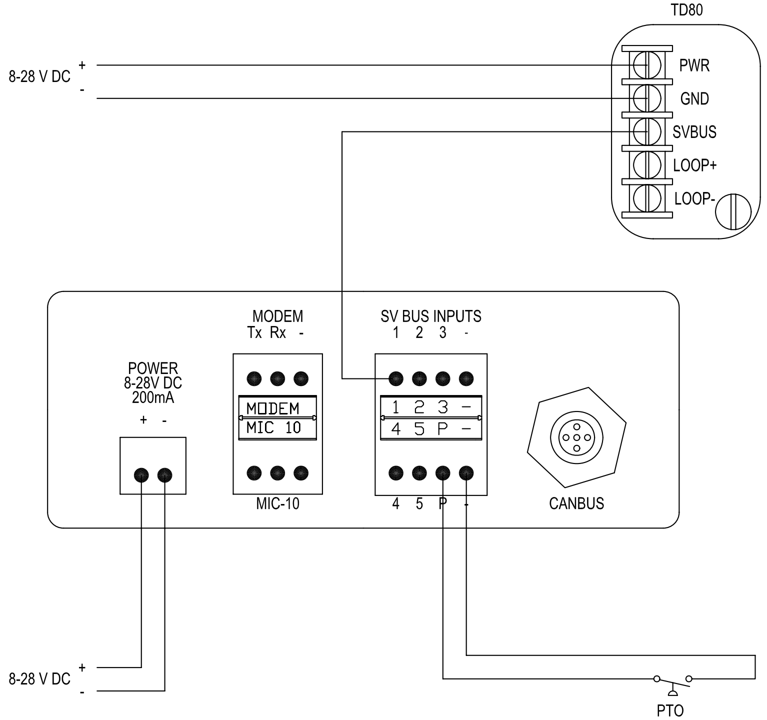
The MIC 10 requires a supply voltage of 8-28V DC. Power needs to be connected to the + and – terminals of the 'power' terminal block. If the wire polarity is reversed it will not damage the unit but the unit will not function.

SV bus

The SV bus inputs should be tied directly to the SV bus signal lines going from the TD80 unit to the Finch 5332 display.

PTO

The Power Take Off (PTO) status terminal should be connected to the existing PTO control. When the PTO terminal, labeled as ‘P’ on the ‘SV BUS’ terminal block, is switched to ground ,labeled ‘ - ‘, on the ‘SV BUS’ terminal block the MIC 10 will begin transmitting level data from TD80 units connected to it to an external modem via its RS232 output port.



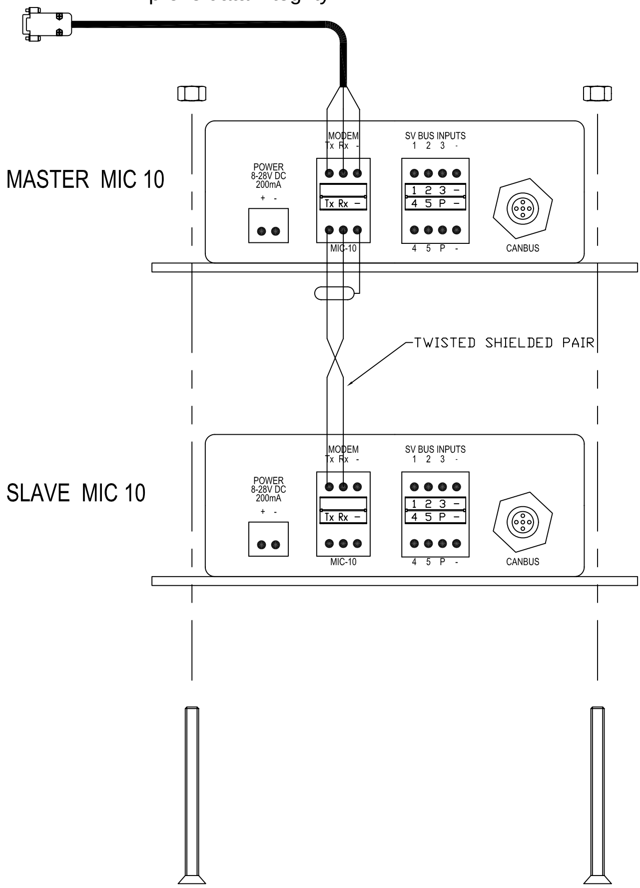
INSTALLATION WIRING PICTORIAL

MIC 10 to Modem

The cable provided can be used to connect the MIC 10 to the RS232 port on a modem. The connector can be plugged directly into the modem. The free wires are connected to the top row of terminals on the ‘MODEM’ terminal block. The white wire gets connected to the Tx terminal and the black wire gets connected to the Rx terminal on the MIC 10. Shield gets connected to the ‘ - ‘ terminal.

MIC 10 to MIC 10

Communication lines going from the MIC 10 to an adjacent MIC 10 unit or to the modem should use a shielded cable to improve data integrity.



Three wires are required between the ‘MODEM’ terminals on the Slave MIC 10 and the ‘MIC 10' terminals on the Master unit. They are Rx, Tx and Ground. The Tx and Rx wires between the MIC 10 units need to be crossed (ie. the Rx from the slave unit must be connected to the Tx of the master unit and vice versa).

MIC 10 to MTD

TD80 level information can be transmitted to a TD80 Multi-Tank Display (MTD) via the CAN port. For more details on the operation of the TD80 Multi-Tank Display please refer to its operations manual.

3 Operation

When operating the MIC 10 will start transmitting data to a modem immediately after the PTO contacts are closed. Closing the PTO contacts indicates the starting fluid level for a fluid transfer operation. The MIC 10 will then transmit a first package indicating the fluid level at the start of operations. The MIC 10 will continue transmitting data at user set intervals throughout the fluid transfer until the PTO contacts are opened indicating that fluid transfer has stopped. The MIC 10 will then send a final fluid level as soon as the PTO has been opened and it will stop transmitting to the modem.

Power up

When power is applied to the MIC 10 unit, it will complete a startup sequence and the run LED will start flashing about once per second indicating the unit is functioning properly.

The green LED’s labeled 1 to 5 on the front of the MIC 10 indicate when data is being received on there respective SV port from a TD80. They will flash at about 1 second intervals when receiving data.

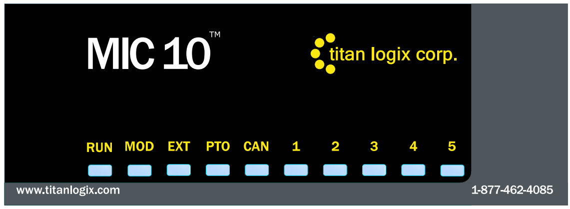
TD80 devices require ten seconds to power up and transmit their first data over the SV bus so there will be a start up delay for the LED’s to start flashing.

A slave MIC 10 will start transmitting TD80 data immediately after power up. A start up delay of about 13 seconds on power up on the use of incoming data should be incorporated to ensure that up to date data has been collected and routed to the master MIC 10 unit.

PTO switch

If the PTO contact of the master MIC 10 unit is open it will sit idle until the PTO switch is closed. Once the PTO switch is closed the amber PTO LED on the front of the MIC 10 will turn on indicating the master MIC 10 has begun broadcasting data.
A powered Slave MIC 10 connected to the ‘Ext Modem’ port on a master MIC 10 will *always* broadcast data the master regardless of the state of its PTO contact.

3.1 Indicators



LED's

There are several LED's on the front of the MIC 10 indicating system activity. The list below describes what each LED indicator represents.

RUN

This LED indicates system status. A flashing red LED indicates a functioning CPU.

MOD

This amber LED indicates the current state of the RS232 line coming from the upstream modem.

EXT

This amber LED indicates the current state of the RS232 lines coming from Slave devices.

PTO

The PTO's amber LED indicates the current state of the PTO input.

CAN

This amber LED indicates the current state of the CAN port.

1…5

These green LED's indicate the status of SV inputs 1 through 5. They will flash when SV communication is active on their respective input.

3.2 Master MIC 10

Master / Slave

If two MIC 10 units are connected together the one connected directly to the modem will be the master. The MIC 10 will automatically set itself as a Master or a Slave depending upon how it is connected to other MIC 10 devices.

A master MIC 10 will transmit data, to the modem, that it has obtained from the TD80 units connected directly to it as well as the data from other MIC 10 units. The updates will be sent to the modem at an interval rate defined by the Update Rate Jumpers. If the PTO switch connected to a master is open then the modem will not be updated.

3.3 Slave MIC 10

A Slave MIC 10 transmits the data, to the master MIC 10, that it has collected from the SV bus of connected TD80 inputs once a second.

3.4 Communications

The Master MIC 10 will send all the error, alarm, and level information on a change of state of the master PTO input. If the PTO contacts are closed data will be sent at an interval as defined by the Update Rate Jumper setting on the Master unit.

Modem Communication

When a MIC 10 is transmitting or receiving data from the modem the amber “MOD’ LED will flash indicating that communications are active on the modem port.

MIC 10 to MIC 10 Com.

When a MIC 10 is transmitting or receiving data from a slave MIC 10 the amber “EXT” LED will flash indicating that communications are active on the external MIC 10 port.

CAN port

When a MIC 10 is actively communicating with a TD80 MTD the amber “CAN” LED will flash indicating that communications are active on the CAN port.

Data Package

Technical information detailing the data package sent to a modem or from one MIC 10 to another can be found in section 6, MIC 10 Communications Protocol, of this manual.

Technical information describing SV bus communications used by the TD80 can be found in the SV Bus Communications section of this manual.

4 Troubleshooting

Problem	Possible Solution
No LEDs light up on the unit	Check the polarity and voltage of the input power, if reverse polarity is applied the unit will not function. The MIC-10 requires 8V – 28V input voltage. The RUN LED will flash regularly if power is applied properly.
No communication via the serial port	1. Make sure the PTO pin on the MIC-10 is connected to GND ‘ - ‘ to enable outgoing communication. 2. Ensure the serial cable to the modem/terminal is connected to the ‘MODEM’ serial port on the MIC 10. 3. Check that the Rx pin on the MIC 10 is connected to the Tx pin on the modem/terminal and the Tx pin on the MIC 10 is connected to the Rx pin on the modem/terminal 4. Ensure the modem is set for 9600 baud 8N1
Receiving data from MIC-10 but no level data	Check the SV bus connection from the TD80 to the MIC 10. Check power to the TD80. If the MIC 10 is receiving data from the TD80 the corresponding LED will flash on the front of the MIC 10.

5 Specifications

Power	8-28 V DC
Current Consumption:	25mA at 12VDC 20mA at 24VDC
Ambient Temperature Range:	-40°C to +65°C
Humidity	5% - 95% (Non Condensing)
Communications	SV port RS232 port CAN port

6 MIC 10 Communications Protocol

Communications from the MIC 10 to a modem takes place at 9600 baud, no parity, one start bit and one stop bit. The Master MIC 10 will send all the error, alarm, and level information on a change of state of any PTO input or if the PTO is ON at an interval as defined by the Update Rate Jumper setting on the Master unit. An active input is defined as an input that has its PTO switch on and is communicating.

Data Packet

The basic packet is broadcasted at intervals it consists of a preamble, one or more data records, and a CRC. One record contains all the information from one active input. Therefore, a packet will contain as many records as there are active inputs. If a slave MIC 10 is connected to a master then the SV address for TD80 units connected to the slave will be increased by 5 before being reported to the modem by the master. The format of a data packet is described below.

<table>
<tr><th>Byte Offset</th><th>Value</th><th>Description</th></tr>
<tr><td>0x01</td><td>‘$’
0x24</td><td>First byte of preamble, synchronization byte.</td></tr>
<tr><td>0x02</td><td>‘L’
0x4C</td><td>Second byte of preamble, modem load command.</td></tr>
<tr><td>0x03</td><td>0x15</td><td>Record Header (of first record)
<table>
<tr><th>Field</th><th>Value</th><th>Meaning</th></tr>
<tr><td rowspan="2">Bit 0</td><td>0</td><td>This record is invalid, discard data.</td></tr>
<tr><td>1</td><td>This record is valid.</td></tr>
<tr><td rowspan="2">Bit 1</td><td>0</td><td>This is not the last record in the packet.</td></tr>
<tr><td>1</td><td>This is the last record in the packet.</td></tr>
<tr><td rowspan="2">Bits 7:2</td><td>5</td><td>This record contains information from a TD80 (normal, default record type).</td></tr>
<tr><td>3</td><td>This record contains a command from the Modem to the MIC 10.</td></tr>
</table></td></tr>
<tr><td>0x04</td><td>0x00</td><td>Device number (of first record)
Identifies which device this record’s information is from.
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>0</td><td>SV input #1 (local to Master MIC 10).</td></tr>
<tr><td>1</td><td>SV input #2 (local to Master MIC 10).</td></tr>
<tr><td>...</td><td></td></tr>
<tr><td>5</td><td>SV input #6 (local input #1 on Slave MIC 10).</td></tr>
</table></td></tr>
</table>

		6 — SV input #7(local input #2 on Slave MIC 10). ... 9 — Highest possible value, input #5 on Slave MIC 10.	
0x05	0x00	Error Message (of first record). See the **TD80 Error Messages** for error message descriptions.	
0x06	0x04	Alarm Messages See the **TD80 Alarm Messages** for alarm message descriptions.	
0x07	0x00	MSB of level (of first record)	
0x08	0x39	LSB of level (of first record)	
0x09	0x07	MSB of HH Level (of first record)	
0x0A	0xD0	LSB of HH level (of first record)	
0x0B	0x17	Record Header (of second record)	
0x0C	0x03	Device Number (of second record)	
0x0D	0x00	Error Code (of second record)	
0x0E	0x01	Alarm Code (of second record)	
0x0F	0x0A	MSB of level (of second record)	
0x10	0x6B	LSB of level (of second record)	
0x11	0x07	MSB of HH Level (of second record)	
0x12	0xD0	LSB of HH level (of second record)	
0x13	0xD2	MSB of CRC	"Modbus" CRC-16 Covers all bytes of packet.
0x14	0xCD	LSB of CRC	

Packet Length

If there are more or less active inputs, the packet will contain more or less records. If there are no active records, the packet will only contain one record and in the record header it will be flagged as invalid.

CRC

This CRC table is used to test packet validity crc_rt_int gets initialized to FFFF hex before any data string is processed. After one byte is processed crc_rt_int will contain the result of the last crc16c function call. cdat is the byte that contains the 8 bit value being processed.

```
unsigned int code crc16a[]=
{
0000000,   0140301,    0140601,   0000500,
0141401,   0001700,    0001200,   0141101,
0143001,   0003300,    0003600,   0143501,
0002400,   0142701,    0142201,   0002100,
};


unsigned int code crc16b[]=
{
0000000,   0146001,   0154001,   0012000,
0170001,   0036000,   0024000,   0162001,
```

```
0120001,   0066000,   074000,    0132001,
0050000,   0116001,   0104001,   0042000,
};

unsigned int crc16c(unsigned char cdat, unsigned int
crc_rt_int) using 2
{
register unsigned int crc_temp;
crc_temp = cdat ^ crc_rt_int;
crc_rt_int = (crc_rt_int / 256) ^ crc16a[(crc_temp & 0017)] ^
crc16b[(crc_temp & 0360) / 16];
return(crc_rt_int);
}
```

Example

The example packet described below would be sent from the MIC 10 to the Modem through the Modem Port in the following scenario:

There are 2 active inputs, SV #1 and SV#4

- SV #1
  - Fluid level is 57
  - HH Level is 2000
  - The “Too Low” alarm is active
- SV#4
  - Fluid level is 2667
  - HH Level is 2000
  - The “HH” alarm is active

MIC 10 Message

24 4C 15 01 00 40 00 39 07 D0 17 04 00 00 0A 6B 07 D0 56 8A

24	synchronization byte		$
4C	modem load command		L
15	Record Header (First Record)		
	Bit	value	meaning
	0	1	valid packet,
	1	0	Not last record
	2:7	5	TD80 Data
01	Device Number	SV Input #2	
00	Error Code	no error	
40	Alarm Code	bit 6	2Lo Alarm
00	Fluid Level MSB	0x0039 = 57	
39	Fluid Level LSB	0x0039 = 57	
07	HH Level MSB	0x07D0 = 2000	
D0	HH Level LSB	0x07D0 = 2000	
17	Record Header (Second Record)		
	Bit	value	meaning
	0	1	valid packet,
	1	1	this is the last record
	2:7	5	TD80 Data
04	Device Number	SV Input #5	
00	Error Code	no error	
00	Alarm Code	bit 0	HH Alarm

0A	Fluid Level MSB	0x0A6B = 2667
6B	Fluid Level LSB	0x0A6B = 2667
07	HH Level MSB	0x07D0 = 2000
D0	HH Level LSB	0x07D0 = 2000
56	CRC MSB	
8A	CRC LSB	

Commands to MIC 10

The MIC 10 also has the ability to receive commands from the modem.
The "Send All Records" command instructs the MIC 10 to send a packet containing records from all valid inputs (exactly the same packet this automatically broadcast at intervals). This command is useful when the broadcast rate has been set very slow (once per hour for instance) and data is needed immediately.

The "Send Single Record" command instructs the MIC 10 to send a packet containing exactly one record with information from a specific device, e.g. "Send data from SV input #7 now". If information from an inactive input is requested, a packet is returned with one invalid record.

The MIC 10 is not able to relay commands to or program strap table in a TD80.

Byte Offset	Value	Description	
0x01	'$' 0x24	First byte of preamble, synchronization byte.	
0x02	'L' 0x4C	Second byte of preamble, MIC 10 load command.	
0x03	0x0F	Command Record Header.	
0x04	0x??	Opcode 0x91 = "Send All Records" 0x90 = "Send Single Record" 0x6B="Send firmware versions"	
0x05	0x??	Command Data Only valid for "Send Single Record" command. This is the zero-based index of the device that is requested. Must be '00' otherwise.	
0x06	0x00	Future	
0x07	0x00	Future	
0x08	0x00	Future	
0x09	0x00	Future	
0x0A	0x00	Future	
0x0B	0x??	MSB of CRC.	"Modbus" CRC-16 Covers all bytes of packet.
0x0C	0x??	LSB of CRC.	

Example

The example packet described below would be sent from the modem to the MIC 10 through the Modem Port in the following scenario:

Send all Records

Modem Message 24 4C 0F 91 00 00 00 00 00 00 0B 07

24	synchronization byte		'$'
4C	modem load command		'L'
0F	Command Record Header		
	Bit	value	meaning
	0	1	valid packet
	1	1	last record
	2:7	3	command to MIC 10
91	Command		Send all Records
00	command data		0
00	future		
00	future		
00	future		
00	future		
00	future		
0B	CRC MSB		
07	CRC LSB		

Firmware Rev. The MIC 10 if requested will provide the revision number of its current firmware in all MIC 10's connected to the system.

Sample Response 24 4C 0F 6E 00 01 00 01 FF FF 4D F6

24	'$'		
4C	'L'		
0F	Command Record Header		
	Bit	value	meaning
	0	1	valid packet
	1	1	last record
	2:7	3	command to MIC 10
6E	Reply		Firmware Versions
00	1st MIC 10 LSB		
01	1st MIC 10 MSB		Version 1
00	2nd MIC 10 LSB		
01	2nd MIC 10 MSB		Version 1
FF	no MIC 10		
FF	no MIC 10		
4D	CRC MSB		
F6	CRC LSB		

The revision orders appear in the same order in which the MIC 10's are wired together. It may take up to 10 seconds for the MIC 10 to respond with version information.

CAN port The MIC 10 hardware has been designed to support communications via a CAN port. At this time the CAN port is used only for proprietary communications between the MIC 10 and a Titan Logix Corp. TD80 Multi-Tank Display.

7 SV Bus Communications

The primary function of the MIC 10 unit is that if an interface device between TD80's and a modem. It collects data from the TD80 via its SV bus. The Finch 5332 display when connected also uses the SV bus. Within the first 2 seconds of power up the Finch may send a calibration command to the TD80. The TD80 then listens for a calibration command. If this calibration command is not sent then the TD80 starts transmitting either level information or an error information.

The SV bus inputs on the MIC 10 unit are read only. They will always be passive to the communications taking place between the TD80 and the display unit.

Protocol

Each data field of the SV bus message contains one or more bytes of ASCII binary code. Each byte contains 1 start bit, 8 data bits, 1 stop bit, and no parity. The data transmission rate is 1200 bits per second. The length of the data fields, "r" frames, is fixed at 10 bytes.

Byte	Example	Description
0	'r'	Lower case r for reply to a READ command (which was not sent by the Finch Display since the transmitter sends data automatically). This is for compatibility with older systems.
1	0x01	A constant value of 0x01, which is the gauge number (in hex format).
2	 0x0B 0x13	Register number where the transmitted value (Byte 3&4) was read from (in hex format). Register 0x0B = level data Register 0x13 = Error data
3		Level data or Error data depending upon flag set in byte 2. Level Data (MSB) 16 bit tank level sent as 2 hex bytes (max. number is 9999) Error Data - see TD80 Error Messages
4		Level data or Error data depending upon flag set in byte 2. Level Data (LSB) Error Data - see TD80 Error Messages
5		One byte of 8 binary bits coding alarm relay settings and various flags: Bit Alarm (LSB) 0 High High 1 Spare 2 Spare

	0x40	3 Spare 4 Spare 5 Spare 6 First 6” indicator (2Lo) 7 Overfill indicator (Spill)
6		High High (HH) level (MSB). 16 bit High High tank level sent as 2 hex bites. This will be used by the Finch Display for the Max. tank volume. The operator programmed Fill point cannot exceed this value.
7		High High (HH) level (LSB).
8	0x2A	MSB of 16 bit checksum of this data packet. Checksum is calculated on all data as 2’s compliment of the actual sum (0-sum).
9	0xDC	LSB of 16 bit checksum of this data packet.

TD80 Alarm Message

One byte of 8 binary bits coding alarm relay settings and various flags:

Bit	Alarm
0	High High
1	Spare
2	Spare
3	Spare
4	Spare
5	Spare
6	Level to low to detect (2Lo)
7	Overfill indicator (SPILL)

TD80 Error Messages

The TD80 Type T Level Transmitter is constantly performing error checking routines on the system. Any errors which are detected are coded and sent to the Finch display which shows them in the form of E_xx (where xx is the two digit error code). These errors can assist in diagnosing any problems which may occur in the use of the system.

The error codes, their meanings, and possible solutions are as follows:

Error	Meaning	Possible Solution
0x00	Can't auto range (Could not measure level)	Ensure the probe is not bowed or bent in any way. Move the probe to a location where there is less turbulence, or use a stilling well around the probe. Possible transmitter internal error.
0x01	Too many samples rejected. (Too much turbulence)	Move the probe to a location with less turbulence. Try using a stilling well around the probe.
0x02	Timer 1 count is too large.	Move the probe to a location with less turbulence. Try using a stilling well around the probe.
0x04	Timer 1 count is too large.	Internal transmitter error.
0x10	Time out between captures.	Internal transmitter error.
0x20	No fiducial detected.	Internal transmitter error Possible faulty probe Possible turbulence in the tank. If the problem persists, try using a stilling well around the probe.
0x40	Watchdog reset.	Internal transmitter error. If problem persists, contact your distributor.
0x80	HH Alarm level set too close to Spill Alarm level	
0x81	Alarms set, No strapping table. Transmitter is not suitable for use.	Contact your distributor for programming of the strapping table.
0x83 or 0x84	An error was detected in the strapping table during operation.	Restart the system. If the problem persists, contact your distributor.

Error codes 01, 02, 04, 10, 20, and 40 can be combined if there is more the one error at a time, for example E_26 would be E_02 + E_04 + E20. E_80 to E_84 will not be combined with any other error.