# MM4Z Module User Guide

**Revision 1.5**

# 1.0 Package and Case Information

## 1.1 Package Overview

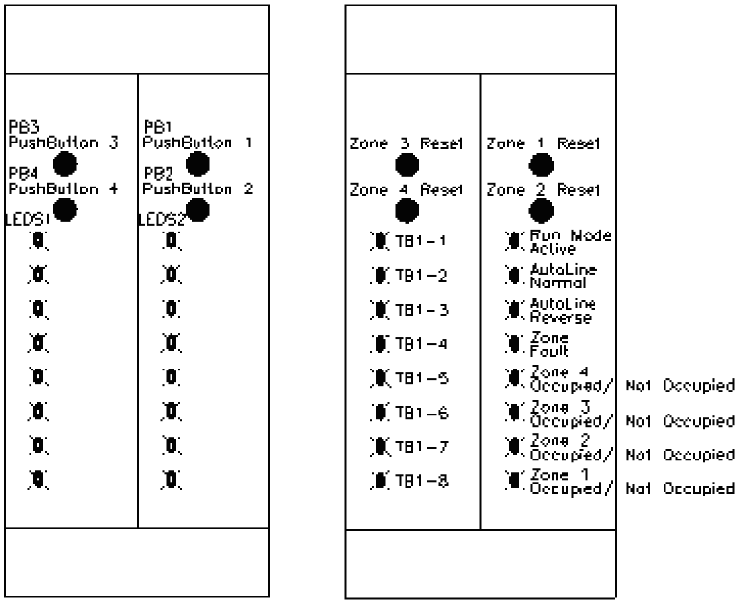


**Figure 1-0 Package top view**

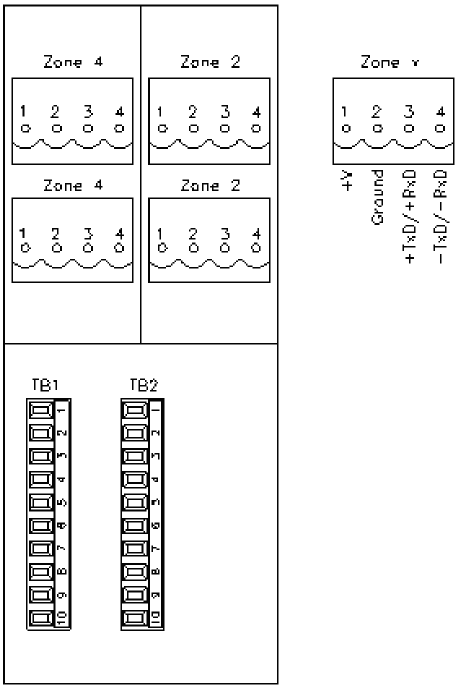


**Figure 1-1 Package side view**

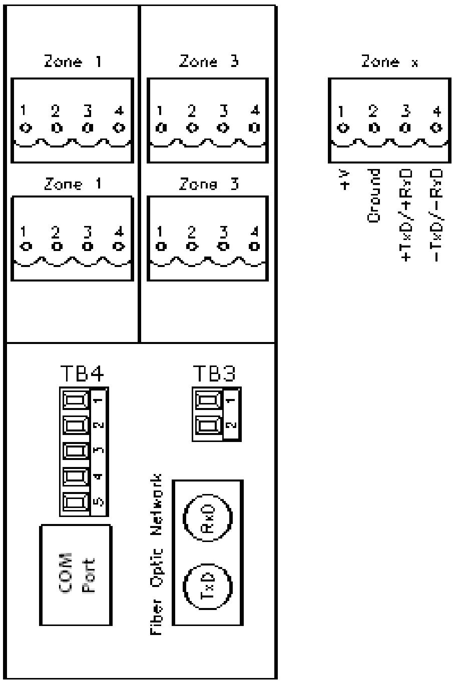


**Figure 1-2 Package side view**

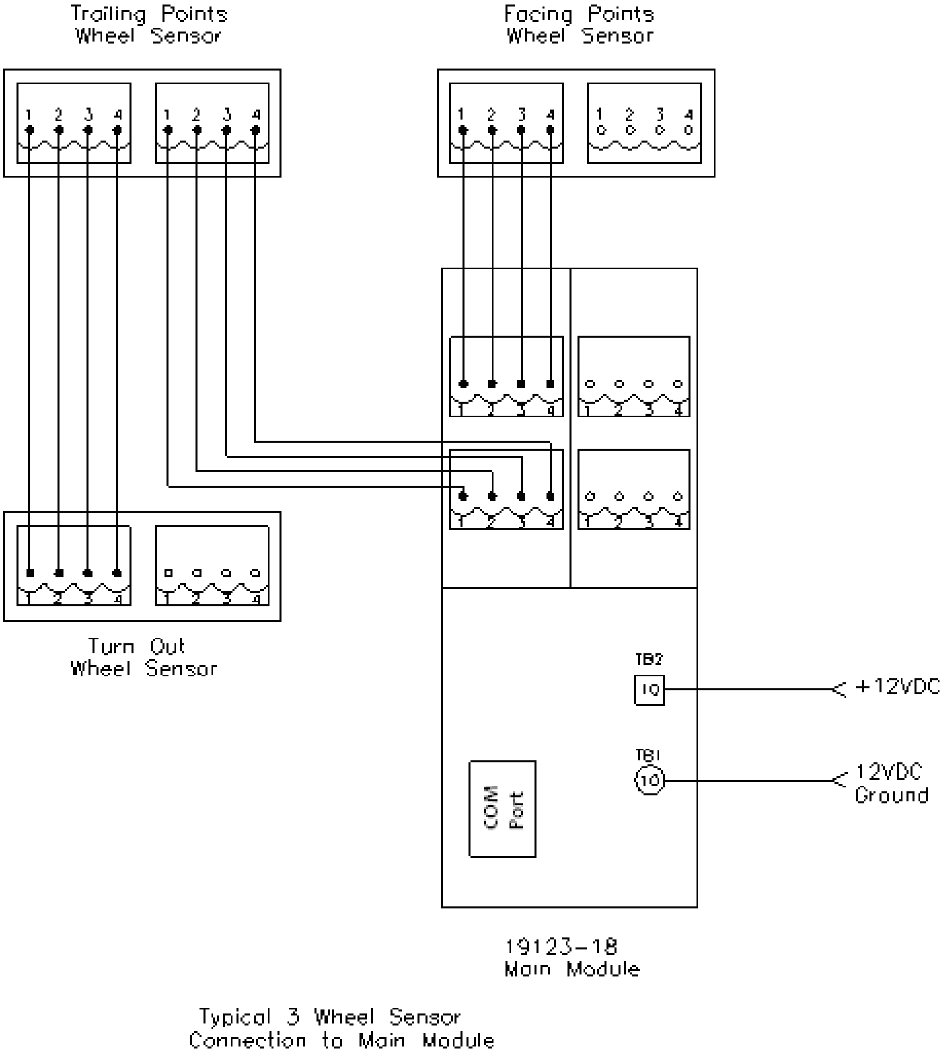


**Figure 1-3 Module/Wheel Sensor Connection Overview**

## 1.2 LEDs and Pushbuttons

### 1.2.0 LEDs

LEDs are grouped into two separate LED strips. LEDS1 indicate input pins, while LEDS2 indicate output pins.

**Table 1-0 LEDS1 Definition**

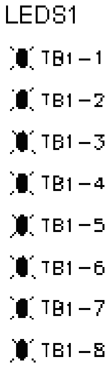
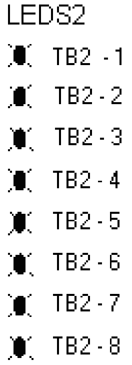
| | | Position | Description | Definition |
|---|---|---|---|---|
| **Input Indicators (LEDS1)** |  LEDS1<br>TB1-1<br>TB1-2<br>TB1-3<br>TB1-4<br>TB1-5<br>TB1-6<br>TB1-7<br>TB1-8<br>**Figure 1-4 LEDS1 (Input Indicators)** | LED8 | Terminal Block 1 Pin 1 | On – Power applied to pin |
| | | | | Off – Power not applied to pin |
| | | LED7 | Terminal Block 1 Pin 2 | On – Power applied to pin |
| | | | | Off – Power not applied to pin |
| | | LED6 | Terminal Block 1 Pin 3 | On – Power applied to pin |
| | | | | Off – Power not applied to pin |
| | | LED5 | Terminal Block 1 Pin 4 | On – Power applied to pin |
| | | | | Off – Power not applied to pin |
| | | LED4 | Terminal Block 1 Pin 5 | On – Power applied to pin |
| | | | | Off – Power not applied to pin |
| | | LED3 | Terminal Block 1 Pin 6 | On – Power applied to pin |
| | | | | Off – Power not applied to pin |
| | | LED2 | Terminal Block 1 Pin 7 | On – Power applied to pin |
| | | | | Off – Power not applied to pin |
| | | LED1 | Terminal Block 1 Pin 8 | On – Power applied to pin |
| | | | | Off – Power not applied to pin |

**Table1-1 LEDS2 Definition**

| | | Position | Description | Definition |
|---|---|---|---|---|
| **Output Indicators (LEDS2)** |  LEDS2<br>TB2 -1<br>TB2-2<br>TB2-3<br>TB2-4<br>TB2-5<br>TB2-6<br>TB2-7<br>TB2-8<br>**Figure 1-5 LEDS2 (Output Indicators)** | LED8 | Run Mode Active | On – Run Mode Active |
| | | | | Off – Run mode Not Active |
| | | LED7 | Auto Line Normal | On – Auto Line Normal in progress |
| | | | | Off – Auto Line Normal not in progress |
| | | LED6 | Auto Line Reverse | On – Auto Line Reverse in progress |
| | | | | Off – Auto Line Reverse not in progress |
| | | LED5 | Zone Fault | On – Zone fault has occurred |
| | | | | Off – Zone fault has not occurred |
| | | LED4 | Zone 4 Status | On – Zone Not Occupied |
| | | | | Off – Zone Occupied |
| | | LED3 | Zone 3 Status | On – Zone not Occupied |
| | | | | Off – Zone Occupied |
| | | LED2 | Zone 2 Status | On – Zone Not Occupied |
| | | | | Off - Zone Occupied |
| | | LED1 | Zone 1 Status | On – Zone Not Occupied |
| | | | | Off – Zone Occupied |

### 1.2.1 Pushbuttons

Four pushbuttons are provided for individual zone resets. Pushbuttons have three functions.

Function 1: During Run Mode Wait Sequence (See section 4.2.0), Pushbuttons are used to activate individual zones and enter Run Mode Polling Sequence (See Section 4.2.1). Pushbutton must stay depressed for at least 300 milliseconds to activate zone.

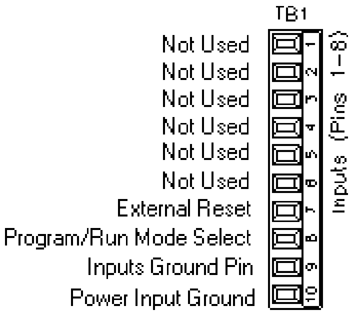
Function 2: After a zone is already activated, pushbuttons are used to reset individual Zones. Pushbutton must stay depressed for at least 300 milliseconds to reset Zone.

Function 3: If Zone 1 and Zone 2 pushbuttons are held down at power up, default network configuration will be applied to the MM4Z Ethernet port (IP = 192.168.1.8, Gateway = 192.168.1.1, 8 host bits).
The Module will acknowledge the configuration change by flashing the Run light. This procedure only applies to Firmware version 1.23 and above.

## 1.3 Terminal blocks

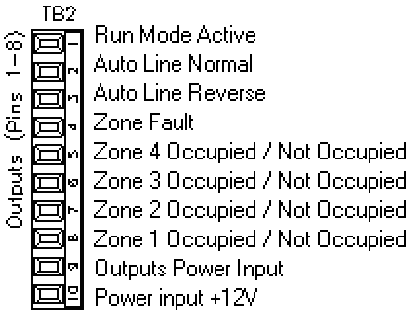
### 1.3.0 Terminal Block 1 (TB1)

**Table 1-2 Terminal Block 1 Pinout**

| | | Pin | Description | Definition |
|---|---|---|---|---|
| **Inputs (TB1)** |   TB1<br>Not Used<br>Not Used<br>Not Used<br>Not Used<br>Not Used<br>Not Used<br>External Reset<br>Program/Run Mode Select<br>Inputs Ground Pin<br>Power Input Ground<br>1<br>2<br>3<br>4<br>5<br>6<br>7<br>8<br>9<br>10<br>Inputs (Pins 1-8) **Figure 1-6 Terminal Block 1 (Inputs)** | 1 | Not Used | |
| | | 2 | Not Used | |
| | | 3 | Not Used | |
| | | 4 | Not Used | |
| | | 5 | Not Used | |
| | | 6 | Not Used | |
| | | 7 | External Reset | Pulled High to reset |
| | | 8 | Program/Run Mode Select | Pulled High at power up - Enter program mode<br>Else, - Enter run mode |
| | | 9 | Input Pins Ground Input | Ground reference for input pins |
| | | 10 | Power Input Ground | Ground reference for MM4Z Module |

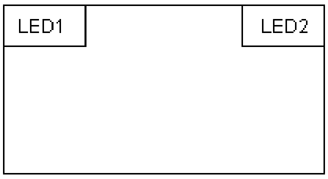
### 1.3.1 Terminal Block 2 (TB2)

**Table 1-3 Terminal Block 2 Pinout**

| | | Pin | Description | Definition |
|---|---|---|---|---|
| **Outputs (TB2)** |   TB2<br>Run Mode Active<br>Auto Line Normal<br>Auto Line Reverse<br>Zone Fault<br>Zone 4 Occupied / Not Occupied<br>Zone 3 Occupied / Not Occupied<br>Zone 2 Occupied / Not Occupied<br>Zone 1 Occupied / Not Occupied<br>Outputs Power Input<br>Power input +12V<br>1<br>2<br>3<br>4<br>5<br>6<br>7<br>8<br>9<br>10<br>Outputs (Pins 1-8) **Figure 1-7 Terminal Block 2 (Outputs)** | 1 | Run Mode active | High – Run Mode Active |
| | | | | Low - Run Mode Not Active |
| | | 2 | Auto Line Normal | High – Auto Line Normal in progress |
| | | | | Low – Auto Line Normal Not in progress |
| | | 3 | Auto Line Reverse | High – Auto Line Reverse in progress |
| | | | | Low – Auto Line Reverse Not in progress |
| | | 4 | Zone Fault | High – Zone Fault Occurred |
| | | | | Low – No Zone Fault Occurred |
| | | 5 | Zone 4 Occupied/Not Occupied | High – Zone Unoccupied |
| | | | | Low – Zone Occupied |
| | | 6 | Zone 3 Occupied/Not Occupied | High – Zone Unoccupied |
| | | | | Low – Zone Occupied |
| | | 7 | Zone 2 Occupied/Not Occupied | High – Zone Unoccupied |
| | | | | Low – Zone Occupied |
| | | 8 | Zone 1 Occupied/No Occupied | High – Zone Unoccupied |
| | | | | Low – Zone Occupied |
| | | 9 | Output Pins +V reference | +V reference for output pins. Output Pins 1-8 will be pulled up +V reference when asserted High |
| | | 10 | Power Input +12V | Power input for MM4Z Module |

## 1.4 Ethernet Port

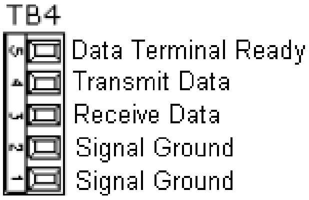
### Table 1-4 Ethernet Port

<table>
<tr><td rowspan="5"><b>Ethernet Port</b></td><td rowspan="5">

<b>Figure 1-8 Ethernet Port Front View</b></td><td colspan="4">Ethernet Network Interface</td></tr>
<tr><th colspan="2">LED1</th><th colspan="2">LED2</th></tr>
<tr><th>Color</th><th>Meaning</th><th>Color</th><th>Meaning</th></tr>
<tr><td>Off</td><td>No Link</td><td>Off</td><td>No Activity</td></tr>
<tr><td>Amber<br>Green</td><td>10 Mbps<br>100 Mbps</td><td>Amber<br>Green</td><td>Half-Duplex<br>Full-Duplex</td></tr>
</table>

## 1.5 Programming port

### Table 1-5 Programming Port

<table>
<tr><td rowspan="6"><b>Programming Input TB4</b></td><td rowspan="6">

<b>Figure 1 - 9 Programming Port</b></td><th>Pin</th><th>Description</th><th>Definition</th></tr>
<tr><td>1</td><td>Data Terminal Ready</td><td></td></tr>
<tr><td>2</td><td>Transmit Data</td><td>Data Transmitted by the Module</td></tr>
<tr><td>3</td><td>Receive Data</td><td>Data Received by the Module</td></tr>
<tr><td>4</td><td>Signal Ground</td><td>Ground Reference for Rec./Trans. Data</td></tr>
<tr><td>5</td><td>Signal Ground</td><td>Ground Reference for Rec./Trans. Data</td></tr>
</table>

## 2.0 Communication and Function Codes

When communicating with the MM4Z Module, Module ID 0xFF is used as a universal address. All connected Modules will answer to Module ID 0xFF.

## 2.1 Ethernet Send/Receive Format

### 2.1.0 Ethernet Send Message Format

| Transaction ID (2 Bytes) | | Protocol ID (2 bytes) | | Message Length (2 Bytes) | | Unit ID | Function Code | Memory Address (2 Bytes) | | # Bytes Requested (2 Bytes) | | # Data Bytes |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hi Byte | Lo Byte | Hi Byte | Lo Byte | Hi Byte | Lo Byte | | | Lo Byte | Hi Byte | Lo Byte | Hi Byte | |

Transaction identifier – Identifier of Request/Response transaction
(Transaction identifier copied from request to response message)
Protocol Identifier - always 0000
Message length – Number of bytes to follow
Unit identifier – Module ID Number
Function Code –Function Code
Memory Address – Start Address to Read/Write
# Bytes Requested – Number of Bytes to Read/Write
# Data Bytes – Data Bytes

### 2.1.1 Ethernet Receive Message Format

| Transaction ID<br>(2 Bytes) | | Protocol ID<br>(2 Bytes) | | Message<br>Length<br>(2 Bytes) | | Unit ID | Function Code | # Data Bytes<br>(2 Bytes) | | # Data Bytes |
|---|---|---|---|---|---|---|---|---|---|---|
| Hi<br>Byte | Lo<br>Byte | Hi<br>Byte | Lo<br>Byte | Hi<br>Byte | Lo<br>Byte | | | Hi<br>Byte | Lo<br>Byte | |

Transaction identifier – Identifier of Request/Response transaction.
Transaction identifier copied from request to response message.
Protocol Identifier - always 0000
Message length – Number of bytes to follow
Unit identifier – Module ID Number
Function Code –Function Code
# Bytes Requested – Number of Bytes to Read/Write
# Data Bytes – Data Bytes

## 2.2 Response Codes

**Table 2-0 Response Codes**

| Exception Code | Definition |
|---|---|
| 0x01 | Operation Successful |
| 0x02 | Operation Unsuccessful |
| 0x05 | Improper number of bytes |
| 0xFC | Protected Memory Address |
| 0xFE | Operation prohibited during Run Mode Polling Sequence (See Section 4.2.1) |
| 0xFF | CRC Error |

## 2.3 Function Code 03

Function Code 03 is used to read RAM Locations (See Section 3.1). Up to 256 bytes can be read at a time

## 2.4 Function Code 04

Function Code 04 is used to read Event Log memory locations (See Section 3.2). Up to 128 bytes can be read at a time.

## 2.5 Function Code 05

Function Code 05 is used to read wheel sensor parameters through MM4Z hardware.

## 2.6 Function Code 06

Function Code 06 is used to write wheel sensor parameters through MM4Z hardware.

## 2.7 Function Code 0x10

Function Code 0x10 is used to write Sensor ID / Module ID information to RAM memory space.

Function code 0x10 can only be used while MM4Z Module is in Run Mode Wait state (See Section 4.2.0). If attempting to use Function Code 0x10 while Module is in Polling Mode (See Section 4.2.1), Module will return Response Code 0xFE (Operation prohibited during Run Mode Polling Sequence).

## 2.8 Function Code 0x11

Function Code 0x11 is used to execute remote functions. See section 3.6

# 3.0 Memory Organization

## 3.1 RAM

RAM locations are read using function code 03. RAM data space occupies memory locations 0xF000 to 0xF5FF. Up to 256 data bytes of RAM can be read at a time.

### 3.1.0 Sensor/Module ID RAM space

Module/Sensor ID RAM locations range from 0xF000 – 0xF0FF

**Table 3-0 Sensor ID RAM Locations**

| Memory Location | Definition | | Memory Location | Definition |
|---|---|---|---|---|
| 0xF000 - 0xF003 | Zone 1 Sensor / Module 1 | | 0xF080 - 0xF083 | Zone 3 Sensor / Module 1 |
| 0xF004 - 0xF007 | Zone 1 Sensor / Module 2 | | : | Zone 3 Sensor / Module 2 |
| 0xF008 - 0xF00B | Zone 1 Sensor / Module 3 | | : | Zone 3 Sensor / Module 3 |
| 0xF00C - 0xF00F | Zone 1 Sensor / Module 4 | | : | Zone 3 Sensor / Module 4 |
| 0xF010 - 0xF013 | Zone 1 Sensor / Module 5 | | : | Zone 3 Sensor / Module 5 |
| 0xF014 - 0xF017 | Zone 1 Sensor / Module 6 | | : | Zone 3 Sensor / Module 6 |
| 0xF018 - 0xF01B | Zone 1 Sensor / Module 7 | | : | Zone 3 Sensor / Module 7 |
| 0xF01C - 0xF01F | Zone 1 Sensor / Module 8 | | : | Zone 3 Sensor / Module 8 |
| 0xF020 - 0xF023 | Zone 1 Sensor / Module 9 | | : | Zone 3 Sensor / Module 9 |
| 0xF024 - 0xF027 | Zone 1 Sensor / Module 10 | | : | Zone 3 Sensor / Module 10 |
| 0xF028 - 0xF02B | Zone 1 Sensor / Module 11 | | : | Zone 3 Sensor / Module 11 |
| 0xF02C - 0xF02F | Zone 1 Sensor / Module 12 | | : | Zone 3 Sensor / Module 12 |
| 0xF030 - 0xF033 | Zone 1 Sensor / Module 13 | | : | Zone 3 Sensor / Module 13 |
| 0xF034 - 0xF037 | Zone 1 Sensor / Module 14 | | : | Zone 3 Sensor / Module14 |
| 0xF038 - 0xF03B | Zone 1 Sensor / Module 15 | | : | Zone 3 Sensor / Module 15 |
| 0xF03C - 0xF03F | Zone 1 Sensor / Module 16 | | 0xF0BC - 0xF0BF | Zone 3 Sensor / Module 16 |
| | | | | |
| 0xF040 - 0xF043 | Zone 2 Sensor / Module 1 | | 0xF0C0 - 0xF0C3 | Zone 4 Sensor / Module 1 |
| : | Zone 2 Sensor / Module 2 | | : | Zone 4 Sensor / Module 2 |
| : | Zone 2 Sensor / Module 3 | | : | Zone 4 Sensor / Module 3 |
| : | Zone 2 Sensor / Module 4 | | : | Zone 4 Sensor / Module 4 |
| : | Zone 2 Sensor / Module 5 | | : | Zone 4 Sensor / Module 5 |
| : | Zone 2 Sensor / Module 6 | | : | Zone 4 Sensor / Module 6 |
| : | Zone 2 Sensor / Module 7 | | : | Zone 4 Sensor / Module 7 |
| : | Zone 2 Sensor / Module 8 | | : | Zone 4 Sensor / Module 8 |
| : | Zone 2 Sensor / Module 9 | | : | Zone 4 Sensor / Module 9 |
| : | Zone 2 Sensor / Module 10 | | : | Zone 4 Sensor / Module 10 |
| : | Zone 2 Sensor / Module 11 | | : | Zone 4 Sensor / Module 11 |
| : | Zone 2 Sensor / Module 12 | | : | Zone 4 Sensor / Module 12 |
| : | Zone 2 Sensor / Module 13 | | : | Zone 4 Sensor / Module 13 |
| : | Zone 2 Sensor / Module 14 | | : | Zone 4 Sensor / Module 14 |
| : | Zone 2 Sensor / Module 15 | | : | Zone 4 Sensor / Module 15 |
| | Zone 2 Sensor / Module 16 | | 0xF0FC - 0xF0FF | Zone 4 Sensor / Module 16 |

### Table 3-1 Sensor ID Byte Field

| **Location** | 0xF000 | 0xF001 | 0xF002 | 0xF003 |
|---|---|---|---|---|
| **Definition** | Bit Field<br>(See Table 3-2) | Undefined | Module ID | Sensor ID |

### Table 3-2 Sensor ID Bit Field

| Bits | Description | |
|---|---|---|
| 7:0 | Bit 7 | 1 – Count Down |
| | | 0 – Count Up |
| | Bit 6 | 1 – Auto Line Enabled |
| | | 0 – Auto Line Disabled |
| | Bit 5 | 1 – Auto Line Reverse |
| | | 0 – Auto Line Normal |
| | Bit 4 - Undefined | |
| | Bit 3 - Undefined | |
| | Bit 2 - Undefined | |
| | Bit 1 - Undefined | |
| | Bit 0 - Undefined | |

### 3.1.1 Zone Count RAM space

Zone count RAM locations range from 0xF100 – 0xF10F
Zone counts can only be read if start address of query is 0xF100 – 0xF10F

### Table 3-3 Zone Count RAM space

| Memory Location | Definition |
|---|---|
| 0xF100 - 0xF103 | Zone 1 Current Count |
| 0xF104 - 0xF107 | Zone 2 Current Count |
| 0xF108 - 0xF10B | Zone 3 Current Count |
| 0xF10C - 0xF10F | Zone 4 Current Count |

### 3.1.2 Firmware Version / Unit ID Number RAM space

### Table 3-4 Firmware version / Unit ID number memory locations

| Parameter | Address |
|---|---|
| Firmware Version (Integer Portion) | 0xF37D |
| Firmware Version (Fraction Portion) | 0xF37E |
| Unit ID | 0xF37F |

### 3.1.3 Sensor Count RAM space

Sensor Count RAM locations range from 0xF380 – 0xF3FF
Each Sensor Count is 2 Bytes.

### Table 3-5 Sensor Count RAM Locations

| Memory Location | Definition | | Memory Location | Definition |
|---|---|---|---|---|
| 0xF380 - 0xF39F | Zone 1 Sensor Counts | | 0xF3C0 - 0xF3DF | Zone 3 Sensor Counts |
| 0xF3A0 - 0xF3BF | Zone 2 Sensor Counts | | 0xF3E0 - 0xF3FF | Zone 4 Sensor Counts |

**Table 3-6 Sensor Count Byte Field**

| 0xF380 | 0xF381 |
|---|---|
| MSB | LSB |

### 3.1.4 Wheel Tracking RAM Memory Space

Wheel tracking RAM space ranges from memory locations 0xF50F – 0xF58E
Positive Wheel Counts (Wheels In) memory space range from 0xF50F – 0xF54E
Negative Wheel Counts (Wheels Out) memory space range from 0xF54F – 0xF58E
Each Wheel Sensor Wheel Tracking Count register is 4 bytes long
Each Zone tracks 4 individual sensors

**Table 3-7 Wheel Tracking Memory Locations**

| Location | Definition | Location | Definition |
|---|---|---|---|
| 0xF50F – 0xF51E | Wheels In: Zone 1 | 0xF54F – 0xF55E | Wheels Out: Zone 1 |
| 0xF51F – 0xF52E | Wheels In: Zone 2 | 0xF55F – 0xF56E | Wheels Out: Zone 2 |
| 0xF52F – 0xF53E | Wheels In: Zone 3 | 0xF56F – 0xF57E | Wheels Out: Zone 3 |
| 0xF53F – 0xF54E | Wheels In: Zone 4 | 0xF57F – 0xF58E | Wheels Out: Zone 4 |

## 3.2 Event Log Memory

Event log is read using function code 04.
Event log is split into two separate “pages” (Page1: 0x0000 – 0xFFFF, Page2: 0x10000 – 0x1FFFF) See section 6.2 for explanation of event log paging.
Up to 128 data bytes of event log can be read at a time.

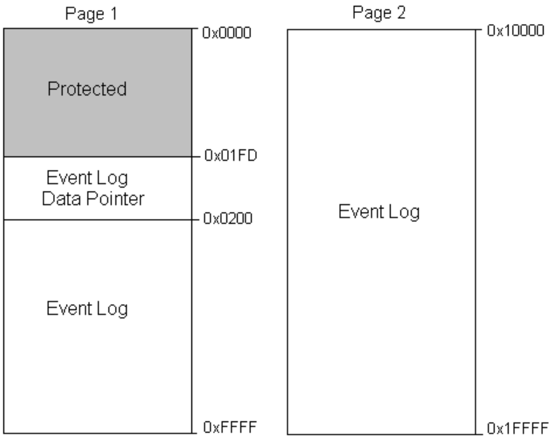


**Figure 3-1 Event Log Memory Map**

**Table 3-8 Event Log Locations**

| Location | Definition |
|---|---|
| 0x01FD | Event Log Page |
| 0x01FE | Event Log Address Hi Byte |
| 0x01FF | Event Log Address Lo Byte |
| 0x0200 - 0x1FFFF | Event Log |

## 3.3 Special Memory Area Using Function Code 0x10

Function code 0x10 can only be used while module is in run mode wait sequence (See Section 4.2.0).

### 3.3.0 Writing Date/Time

Date/Time can be edited by writing 7 bytes to memory location 0x200. Date/Time can be edited in both wait mode and polling mode in Firmware versions 1.23 and above.

**Writing Date/Time Example**

If the current date was: 01-June-2011 (01-06-2011)
And the current time was: 3:20:43 PM = 15:20:43 (*24 hour format*)

**Table 3-9. Date/Time Data Byte Field**

| Byte | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| Definition | Seconds | Minutes | Hour (24 hour format) | Not Used | Day | Month | Year (last two digits) |

Using Table 3-9,
Data byte string = 0x2B 0x14 0x0F 0x00 0x01 0x06 0x0B

**Table 3-10. Ethernet Send Message Format**

| Transaction ID (2 Bytes) | | Protocol ID (2 bytes) | | Message Length (2 Bytes) | | Unit ID | Function Code | Memory Address (2 Bytes) | | # Bytes Requested (2 Bytes) | | Data byte string |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | 0x01 | 0x00 | 0x00 | 0x00 | 0x0D | 0xFF | 0x10 | 0x02 | 0x00 | 0x00 | 0x07 | Data byte string |

Upon successful writing of Date/Time parameters, Module will acknowledge by sending a single return byte 0x01 (Operation Successful)

## 3.4 Function code 0x05

Function code 0x05 is used to read Wheel Sensor parameters over Ethernet or RS485 interface.

### 3.4.0 Supported Wheel Sensor Read parameters

**Table 3-11 Supported Wheel Sensor Read Parameters**

| Address (Hex) | Parameter | Return Format | |
|---|---|---|---|
| | | Hi Byte | Lo Byte |
| 00 | Coil #1 (+) Threshold | xx | 00-FF |
| 01 | Coil #1 (1) Threshold | xx | 00-FF |
| 02 | Coil #2 (+) Threshold | xx | 00-FF |
| 03 | Coil #2 (-) Threshold | xx | 00-FF |
| 04 | Sensitivity | Coil #1 | Coil #2 |
| 05 | Baseline | Coil #1 | Coil #2 |
| AA | Analog Value Coil #1 | xx | 00-FF |
| A5 | Analog Value Coil #2 | xx | 00-FF |
| AD | Counter Contents | 00-FF | 00-FF |
| D5 | ID Number | xx | 00-1F |

### 3.4.1 Wheel Sensor Read Data Format

**Table 3-12 Standard Serial Wheel Sensor Request Data Format**

| STX (fixed) | Sensor ID | Parameter | ETX (fixed) |
|---|---|---|---|
| **02** | xx | See table 3-11 | **03** |

### 3.4.2 Function code 0x05 example

Ethernet communication follows the format outlined in **Section 2.0. Four data bytes are sent in request.**

**Table 3-13 Read From Wheel Sensor Over Ethernet Request Example**

| Transaction ID (2 Bytes) | | Protocol ID (2 bytes) | | Message Length (2 Bytes) | | Unit ID | Function Code | STX | Sensor ID Number | Parameter | ETX |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | 0x01 | 0x00 | 0x00 | 0x00 | 0x06 | 0xFF | 0x05 | 0x02 | xx | See Table 3-11 | 0x03 |

**Table 3-14 Read From Wheel Sensor Over Ethernet Response Example**

| Transaction ID (2 Bytes) | | Protocol ID (2 Bytes) | | Message Length (2 Bytes) | | Unit ID | Function Code | Parameter (Echoed from request) | Data (2 bytes) | | ETX |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | 0x01 | 0x00 | 0x00 | 0x0 0 | 0x06 | 0xFF | 0x5 | See Table 3-11 | Hi byte | Lo byte | 0x03 |

## 3.5 Function Code 0x06

### 3.5.0 Supported Wheel Sensor Write parameters

**Table 3-15 Supported Wheel Sensor Write Parameters**

| Address (Hex) | Parameter | Return Format | |
|---|---|---|---|
| | | Hi Byte | Lo Byte |
| 00 | Coil #1 (+) Threshold | xx | 00-FF |
| 01 | Coil #1 (1) Threshold | xx | 00-FF |
| 02 | Coil #2 (+) Threshold | xx | 00-FF |
| 03 | Coil #2 (-) Threshold | xx | 00-FF |
| AD | Counter Contents | 00-FF | 00-FF |
| D5 | Id Number | xx | 00-1F |

### 3.5.1 Wheel Sensor Write Data Format

**Table 3-16 Standard Wheel Sensor Send Data Format**

| STX (fixed) | Sensor ID | Parameter | 0x2E (fixed) | Data (2 Bytes) | | ETX (fixed) |
|---|---|---|---|---|---|---|
| 02 | xx | See table 3-15 | 0x2E | | | 03 |

### 3.5.2 Function Code 0x06 Example

**Table 3-17 Write to Wheel Sensor Over Ethernet Request Example**

| Transaction ID (2 Bytes) | | Protocol ID (2 bytes) | | Message Length (2 Bytes) | | Unit ID | Function Code | Sensor ID | Parameter | Data (2 Bytes) | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | 0x01 | 0x00 | 0x00 | 0x00 | 0x06 | 0xFF | 0x06 | xx | See Table 3-15 | Hi byte | Lo byte |

**Table 3-18 Write to Wheel Sensor Over Ethernet Response Example**

| Transaction ID (2 Bytes) | | Protocol ID (2 Bytes) | | Message Length (2 Bytes) | | Unit ID | Function Code | Parameter (Echoed from request) | Data (2 bytes) | | ETX |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | 0x01 | 0x00 | 0x00 | 0x00 | 0x06 | 0xFF | 0x06 | See Table 3-15 | Hi byte | Lo byte | 0x03 |

## 3.6 Function Code 0x11

Function code 0x11 is used to execute special functions.

**Table 3-19 Function Addresses**

| Address | Function | Definition |
|---|---|---|
| 0x0000 | Remote Sensor Reset | See Section 3.6.1 |
| 0x0001 | Remote Zone Reset | See Section 3.6.2 |

## 3.6.0 Remote Sensor Reset

**Table 3-20 Remote Sensor Reset Ethernet Request Format**

| Transaction ID (2 Bytes) | | Protocol ID (2 bytes) | | Message Length (2 Bytes) | | Unit ID | Function Code | Address Hi | Address Lo | Data (2 Bytes) | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | 0x01 | 0x00 | 0x00 | 0x00 | 0x06 | 0xFF | 0x11 | 0x00 | 0x00 | 0x00 | Sensor ID # |

## 3.6.1 Remote Zone Reset

**Table 3-21 Remote Zone Reset Request Format**

| Transaction ID (2 Bytes) | | Protocol ID (2 bytes) | | Message Length (2 Bytes) | | Unit ID | Function Code | Address Hi | Address Lo | Data (2 Bytes) | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | 0x01 | 0x00 | 0x00 | 0x00 | 0x06 | 0xFF | 0x11 | 0x00 | 0x01 | 0x00 | See Table 3-22 |

**Table 3-22 Zone Reset bitfield**

| b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|
| x | x | x | x | Reset Zone 4 | Reset Zone 3 | Reset Zone 2 | Reset Zone 1 |

# 4.0 Module Modes and Functions

## 4.1 Program Mode

Program Mode is used to upgrade device firmware.

To enter Program Mode, pin 8 of Terminal Block 1 is pulled high during power up.

While in Programming Mode, the “Update Firmware” option of the MM4Z Module Programmer PC software can be used to update the MM4Z Module firmware (See Section 7.8.2).

To exit Program Mode, the MM4Z Module must be powered up with pin 8 of Terminal Block 1 pulled LOW or left unconnected.

## 4.2 Run Mode

Run Mode is used while executing normal Module functions (i.e. maintaining Zone/Sensor counts)

Run Mode is entered when pin 8 of Terminal Block 1 is pulled low, or left unconnected during module power up.

Entrance into Run mode is verified when LED1 of LEDS2 is on.

### 4.2.0 Run Mode Wait

Upon entering Run Mode, Module will wait for zone activation. Zone activation is performed in one of two ways.

**a. Using Zone Reset Pushbuttons**

Depressing any of the four Zone Reset Pushbuttons for at least 300 milliseconds will active its corresponding zone and send the MM4Z Module into the polling mode (See Section 4.2.1)

**b. Using External Reset**

External reset is provided on TB1 Pin 7. Pulling External Reset up (+12V) for at least 300 milliseconds will activate reset once TB1 Pin 7 returns LOW.

**Upon external reset, the Module will reset / activate all four zones and enter Run Mode Polling.**

While Module is in Run Mode Wait Sequence, writing of Sensor ID/Module ID information using Function Code 0x10 is permitted, and the full event log can be read out of event log memory space.
As soon as any zone is activated, Module immediately enters Run Mode Polling Sequence.

### 4.2.1 Run Mode Polling

When in Run Mode Polling Sequence, all activated Zones will be polled and zone counts will be maintained.

# 5.0 Module Functions

## 5.1 Auto-Line

Auto-line function is used to provide an auto-line pulse on either of the two auto-line outputs (TB2 -1 Auto-Line Normal, TB2-2 Auto-Line Reverse).
Auto-line can be configured on any of the sixteen sensors in zone one.
Auto-Line can only be triggered when a Wheel Sensor configured for auto-line causes the total zone count to increase from zero to a positive number.
Auto-line will not trigger until one wheel has completely passed over the Wheel Sensor configured for auto-line.

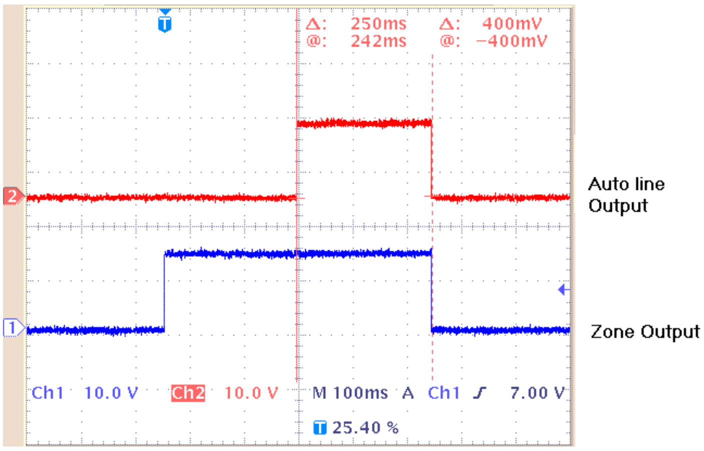


**Figure 5.0 Auto line pulse diagram**

## 5.2 Wheel Compensation

The main function of wheel compensation is to compensate for dragging hoses and hanging chains that might cause a single undesirable count. See D.2 for Wheel Compensation Flow Diagram.

Wheel Compensation has three configurable options

a) Enabled
Compensation will occur for +1/-1 zone count

b) Disabled
Compensation will never trigger

c) Enabled only for -1 count
Compensation will occur only for -1 zone count

### 5.3 Wheel Tracking

Wheel tracking will keep count of wheels entering a zone and wheels leaving a zone for the first four configured sensors in each zone.
Wheel tracking event will be written to the event log when the total zone count returns to zero, when a fault occurs, when a wheel compensation event occurs, or when a zone reset occurs on a previously enabled zone.
See section 3.1.4 for Wheel Tracking RAM Memory Space

### 5.4 Input Logging

Inputs 1 – Inputs 6 are available for input logging.  If Input state changes and holds for at least 250 milliseconds, event 0x000F will be written to the event log recording the current state of the changed input.

## 6.0 Event Log

The Event Log occupies memory locations 0x0200 – 0x1FFFF (See Section 3.2). Each event is comprised of 16 Bytes. Event log can store up to 8160 events at a time.

## 6.1 Event Log Data Pointer

The event log data pointer is 3 bytes occupying event log memory locations 0x01FD – 0x01FF. Event log data pointer always points to the next memory location to be filled by an event.
After writing an event to memory location 0x1FFF0, event log data pointer will rollover and point at memory location 0x0200

**Table 6-0 Event Log Data Pointer Byte Field**

| Location | Definition |
|---|---|
| 0x01FD | Event Log Page |
| 0x01FE | Event Log Address Hi Byte |
| 0x01FF | Event Log Address Lo Byte |

Location 0x01FD (Event Log Page) indicates event log memory “page”. See Section 6.2 for explanation of event log paging.

## 6.2 Event Log Paging

The Hi Byte on “number of registers requested” block of the Ethernet TCP Send message is used to specify the event log page to read.

Ethernet TCP Send Message to read Page 1 Address 0x01FD.

0001 0000 0006 30 04 01FD **00**03
↑
Hi Byte (Read Page 1)

Ethernet TCP Send Message to read Page 2 Address 0x01FD.

0001 0000 0006 30 04 01FD **01**03
↑
Hi Byte (Read Page 2)

### 6.2.0 Reading Single Event from Event Log Example

**Step 1: Reading the Event Log Data Pointer**
The Event Log Data Pointer always points to the next memory address to be filled by an incoming event.

Ethernet TCP Send Message:

0001 0000 0006 30 04 01FD 0003

Ethernet TCP Return Message:

0001 0000 0007 30 04 0003 **00 02 10**

**00 02 10** is the Event Log data pointer

**00 –** Current Event Log "page"
**02 –** Event Log Address Hi Byte
**10 –** Event Log Address Lo Byte

**Step 2: Read the Log**

Ethernet TCP Send Message:
0001 0000 0006 30 04 0200 0010

Ethernet TCP Return Message:
0001 0000 0014 30 04 0010 **00 01 39 15 10 01 12 10 25 00 00 00 01 12 FF FF**

From the above Return Message and the information in Section 6.3, the returned event would be:

Event: Power Up Event
Date: 12/01/2010
Time: 10:15:39
Temp: 25 °C
Inputs: 0x00
Outputs: 0x00
Event Specific Information: Firmware Version 1.18

## 6.3 Event Formats

### Table 6-1 Standard Event Format

| Byte | Description |
|---|---|
| 1 | Event Code Hi Byte |
| 2 | Event Code Lo Byte |
| 3 | Seconds |
| 4 | Minutes |
| 5 | Hour |
| 6 | Date |
| 7 | Month |
| 8 | Year |
| 9 | Temp MSB |
| 10 | Temp LSB |
| 11 | Inputs |
| 12 | Outputs |
| 13 | Event Specific Byte 1 |
| 14 | Event Specific Byte 2 |
| 15 | Event Specific Byte 3 |
| 16 | Event Specific Byte 4 |

### Table 6-2 Event Log Inputs bit Field

| Inputs | Bit 7<br>MSb | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0<br>LSb |
|---|---|---|---|---|---|---|---|---|
| | TB1-1 | TB1-2 | TB1-3 | TB1-4 | TB1-5 | TB1-6 | TB1-7 | TB1-8 |

**1 –Input/Output state high**

**0 –Input/Output state low or floating**

### Table 6-3 Event Log Outputs bit Field

| Outputs | Bit 7<br>MSb | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0<br>LSb |
|---|---|---|---|---|---|---|---|---|
| | TB2-1 | TB2-2 | TB2-3 | TB2-4 | TB2-5 | TB2-6 | TB2-7 | TB2-8 |

### Table 6-4 Event Time/Date/Temp bit Field

| Field | Bit 7<br>MSb | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0<br>LSb | Range |
|---|---|---|---|---|---|---|---|---|---|
| Seconds | 0 | 10 Seconds | | | Seconds | | | | 0x00 – 0x59 |
| Minutes | 0 | 10 Minutes | | | Minutes | | | | 0x00 – 0x59 |
| Hour | 0 | 0 | 10 Hour | 10 Hour | Hour | | | | 0x00 – 0x23 |
| Date | 0 | 0 | 10 Date | | Date | | | | 0x01 – 0x31 |
| Month | Century | 0 | 0 | 10 Month | Month | | | | 0x01 – 0x12 + Century |
| Year | 10 Year | | | | Year | | | | 0x00 – 0x99 |
| Temp MSB | Sign | Data | Data | Data | Data | Data | Data | Data | - |
| Temp LSB | Data | Data | 0 | 0 | 0 | 0 | 0 | 0 | - |

🟨 - Denotes fixed value

Hour Register is read / written in 24hour format
Century bit is set when year rolls over from 99 to 00

Actual Year = (2000 + (100 * Century bit)) + Year count register.
For example, if year count register equals 10 and century bit is not set, then actual year equals 2010

Temperature is represented by a 10 bit code. The lower 7 bits of the Temp MSB register represent the integer value of the current temperature. The MSb of the Temp MSB register represents the temperature sign (0 – Positive, 1 – Negative). The upper 2 bits of the Temp LSB register represent the fraction portion of the current temperature with a resolution of 0.25°C (0x00 – 0.00°C, 0x40 – 0.25°C, 0x80 = 0.50°C, 0xC0 – 0.75°C).
For example, 0x30 in the Temp MSB register and 0x40 in the Temp LSB register would represent 30.25°C.

### Table 6-5 Baseline Fault bit field

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **Definition** | Coil #1 Sensitivity | Coil #2 Sensitivity | Analog Value Coil #1 | Analog Value Coil #2 | Analog Value - Baseline #1 | Analog Value - Baseline #2 | Not Used | Sensor Unresponsive |

### Table 6-6 Module bit field

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **Definition** | 1 - Wheel Compensation Enabled | 1 – Allow Reset After Fault = True | 1 – Allow only -1 Count Compensation | Not Used | Not Used | Not Used | Not Used | Not Used |
| | 0 – Wheel Compensation Disabled | 0 – Allow Reset After Fault = False | 0 – Allow both +1/-1 Count Compensation | | | | | |

### Table 6-7 Non standard Event Format

| Event ID | 0x8012 | 0x0009 | 0x0030 |
|---|---|---|---|
| **Byte** | **Description** | **Description** | **Description** |
| 1 | Event Code Hi Byte | Event Code Hi Byte | Event Code Hi Byte |
| 2 | Event Code Lo Byte | Event Code Lo Byte | Event Code Lo Byte |
| 3 | Zone Number | Seconds | Zone Number |
| 4 | Sensor ID | Minutes | Bit Field |
| 5 | Coil #1 Sensitivity | Hour | Undefined |
| 6 | Coil #2 Sensitivity | Date | Module ID |
| 7 | Coil #1 Baseline | Month | Sensor ID |
| 8 | Coil #2 Baseline | Year | Wheels In Hi Byte |
| 9 | Coil #1 Analog | Inputs | Wheels In Byte 1 |
| 10 | Coil #2 Analog | Outputs | Wheels In Byte 2 |
| 11 | Counter Hi Byte | Seconds (changed to) | Wheels in Byte 3 |
| 12 | Counter Lo Byte | Minutes (changed to) | Wheels Out Hi Byte |
| 13 | Wheel Sensor Bit Field (See Table 3-2) | Hour (changed to) | Wheels Out Byte 1 |
| 14 | Baseline Fault Bit Field (See Table 6-5) | Date (changed to) | Wheels Out Byte 2 |
| 15 | Not used | Month (changed to) | Wheels Out Byte 3 |
| 16 | Not used | Year (changed to) | Not Used |

### 6.2.0 Event Descriptions

**Event ID: 0x0000**
**Definition: PC/Module Date/Time**
**Description:** This event dose not actually exist in the event log memory space of the MM4Z Module. It reads the current PC and Module date time when event log memory space is read

**Event ID: 0x0001**
**Definition:** Power Up
**Description:** Written to log after initial power up delay
**Event Specific Bytes:** Byte 1 – Firmware Version Integer portion
Byte 2 – Firmware Version Fraction portion

**Event ID: 0x0002**
**Definition:** External Reset
**Description:** Written to the log after external reset pin has returned low after being pulled up for at least 250 milliseconds.
**Event Specific Bytes:** Byte 1 – Firmware Version Integer portion
Byte 2 – Firmware Version Fraction portion

**Event ID: 0x0004**
**Definition:** Count Up from Zero
**Description:** Written to the event log when the total zone count goes from a count of zero to a positive number. Corresponding zone will show occupied status before this event is written to the log. Sensor Count Register will be recorded in event specific bytes 3 and 4. Sensor count will reflect the previous count in the corresponding Wheel Sensor (initial count before total zone count increased from zero)
**Event Specific Bytes**: Byte 1 – Zone Number
Byte 2 – Sensor ID
Byte 3 – Sensor Count Hi Byte
Byte 4 – Sensor Count Lo Byte

**Event ID: 0x0005**
**Definition:** Count Down to Zero
**Description:** Written to the event log when the total zone count decreases from a positive count down to zero. This event is written after the corresponding zone output shows unoccupied status. Sensor Count Register will be recorded in event specific bytes 3 and 4. Sensor count will reflect the current count in the corresponding Wheel Sensor (the count in the Wheel Sensor that caused the total zone count to return to zero)
**Event Specific Bytes**: Byte 1 – Zone Number
Byte 2 – Sensor ID
Byte 3 – Sensor Count Hi Byte
Byte 4 – Sensor Count Lo Byte

**Event ID: 0x0006**
**Definition:** Zone Reset
**Description:** Written to the event log after a zone reset pushbutton has been released after being depressed for at least 300 milliseconds. This event is written to the event log prior to the corresponding zone output showing unoccupied status.
**Event Specific Bytes:** Byte 1 – Zone Number

**Event ID: 0x0007**
**Definition:** Unresponsive Sensor
**Description:** Written to the log after a Wheel Sensor is found to be unresponsive either from framing errors, unresponsive retries, or a combination of the two.
**Event Specific Bytes:** Byte 1 – Zone Number
Byte 2 – Sensor ID

**Event ID: 0x0008**
**Definition:** Wheel Compensation
**Description:** Written to the log after the programmed wheel compensation time has elapsed. This event will be written after the total zone count has been reset to zero, but before the corresponding zone output shows unoccupied status.
Event Specific Bytes: Byte 1 – Zone Number
Byte 2 – Sensor ID
Byte 3 – Programmed Wheel Compensation Time Hi Byte
Byte 4 – Programmed Wheel Compensation Time Lo Byte

$C_{Total}$ = Wheel Compensation Time (Seconds)
$C_{Total} = ((\text{Byte } 3 * 256) + \text{Byte } 4) * 0.005$

**Event ID: 0x0009**
**Definition:** Date/Time Change
**Description:** Written to the log after a new date/time is written to the Module. This event does not conform to the standard event log format (See Table 6-7).

**Event ID: 0x000A**
**Definition:** Negative Count Limit Exceeded
**Description:** Written to the log after the total zone count decreases to or below -2 (Zone Count Register <= 0xFFFFFFF8)
**Event Specific Bytes:** Byte 1 – Zone Number
Byte 2 – Sensor ID
Byte 3 – Sensor Count Hi Byte
Byte 4 – Sensor Count Lo Byte

**Event ID: 0x000B**
**Definition:** Auto-Line
**Description:** Written to the log after auto-line occurs
**Event Specific Bytes:** Byte 1 – Sensor ID that triggered auto-line
Byte 2 – Sensor bit field (See Table 3-2)

**Event ID: 0x000D**
**Definition:** New Wheel Compensation Time
**Description:** Written to the log after a new wheel compensation time is configured
**Event Specific Bytes:** Byte 1 – Previous Time Hi Byte
Byte 2 – Previous Time Lo Byte
Byte 3 – New Time Hi Byte
Byte 4 – New Time Lo Byte

**Event ID: 0x000E**
**Definition:** Function Change
**Description:** Written to the log when Module functions are changed (Wheel Compensation Enabled, Disabled, Allow only -1 count, and Allow reset after fault) See Table 6-6.
**Event Specific Bytes:** Byte 1 – Previous Bit Field
Byte 2 – New Bit Field

**Event ID: 0x000F**
**Definition:** Input State Change
**Description:** Written to the log when the state of input 1 – input 6 changes for at least 250 milliseconds.
**Event Specific Bytes:** Byte 1 – Input That Changed
Byte 2 – Current State of input

**Event ID: 0x0010**
**Definition:** Baseline Error
**Description:** Written to the log when a check of a Wheel Sensor’s baselines fails test. Sensor baselines are checked at zone reset and after a fault.
**Event Specific Bytes:** Byte 1 – Zone Number
Byte 2 – Sensor Number
Byte 3 – Baseline Fault Bit Field (See Table 6-5)

**Event ID: 0x0012**
**Definition:** No Sensors Configured In Zone
**Description:** Written to the log when a zone is reset and there are no sensors configured for that zone.
**Event Specific Bytes:** Byte 1 – Zone Number

**Event ID: 0x0014**
**Definition:** Count Down from Zero
**Description:** Written to the log when the total zone count decreases from zero to a negative count. This event is written after the corresponding zone shows occupied status. Sensor Count Register will be recorded in event specific bytes 3 and 4. Sensor count will reflect the previous count in the corresponding Wheel Sensor (initial count before total zone count decreased from zero)
**Event Specific Bytes:** Byte 1 – Zone Number
Byte 2 – Sensor ID
Byte 3 – Sensor Count Hi Byte
Byte 4 – Sensor Count Lo Byte

**Event ID: 0x0015**
**Definition:** Count Up to Zero
**Description:** Written to the log when the total zone count increases from a negative number up to zero. This event is written to the log after the corresponding zone output shows unoccupied status. Sensor Count Register will be recorded in event specific bytes 3 and 4. Sensor count will reflect the current count in the corresponding Wheel Sensor (The count that caused the total zone count to increase to zero).
**Event Specific Bytes:** Byte 1 – Zone Number
Byte 2 – Sensor ID
Byte 3 – Sensor Count Hi Byte
Byte 4 – Sensor Count Lo Byte

**Event ID: 0x0016**
**Definition:** Count Down Through Zero
**Description:** Written to the log when the total zone count decreases from a positive number to a negative number. Sensor Count Register will be recorded in event specific bytes 3 and 4. Sensor count will reflect the current count in the corresponding Wheel Sensor (The count that caused the total zone count to decrease).
**Event Specific Bytes:** Byte 1 – Zone Number
Byte 2 – Sensor ID
Byte 3 – Sensor Count Hi Byte
Byte 4 – Sensor Count Lo Byte

**Event ID: 0x0017**
**Definition:** Count Up Through Zero
**Description:** Written to the log when the total zone count increases from a negative number to a positive number. Sensor Count Register will be recorded in event specific bytes 3 and 4. Sensor count will reflect the current count in the corresponding Wheel Sensor (The count that caused the zone count to increase).
**Event Specific Bytes:** Byte 1 – Zone Number
Byte 2 – Sensor ID

Byte 3 – Sensor Count Hi Byte
Byte 4 – Sensor Count Lo Byte

**Event ID: 0x0030**
**Definition:** Wheel Tracking
**Description:** This event is written to the log when the total zone count returns to zero, a zone reset occurs, a fault occurs, or wheel compensation event occurs. This event does not conform to the standard event log format (See Table 6-7)**.**

**Event ID: 0x8000**
**Definition:** Critical Fault (Loopback Mismatch)
**Description:** This event is written to the log when the loopback state of the four zone occupied/unoccupied outputs do not match the intended output.
**Event Specific Bytes:** Byte 1 – (0xF0 – Output, 0x0F – Loopback)

**Event ID: 0x8007**
**Definition:** Sensor Communication Data - Frame Error/Unresponsive Retries
**Description:** Written to the log when a frame error or unresponsive retry occurs and communication is reestablished, or if *number of frame errors + unresponsive retries = 20*.
**Event Specific Bytes:** Byte 1 – Zone Number
Byte 2 – Sensor ID
Byte 3 – Number of Frame Errors
Byte 4 – Number of Unresponsive Retries

**Event ID: 0x8012**
**Definition:** Record Sensor Baselines
**Description:** This event records the baseline parameters of each Wheel Sensor. It is written to the log at zone reset and after a fault occurs. This Event does not conform to the standard event log format **(**See Table 6-7).

**Event ID: 0x8013**
**Definition:** Threshold Values Error
**Description:** This event occurs when the Positive Threshold Values of the Coils are not equal to 5 or if the Negative Threshold Values of the Coils do not equal 8.
**Event Specific Bytes:** Byte 1 – Coil #1 (+) Threshold
Byte 2 – Coil #1 (-) Threshold
Byte 3 – Coil #2 (+) Threshold
Byte 4 – Coil #2 (-) Threshold

## 7.0 MMZ Module Programmer PC Software

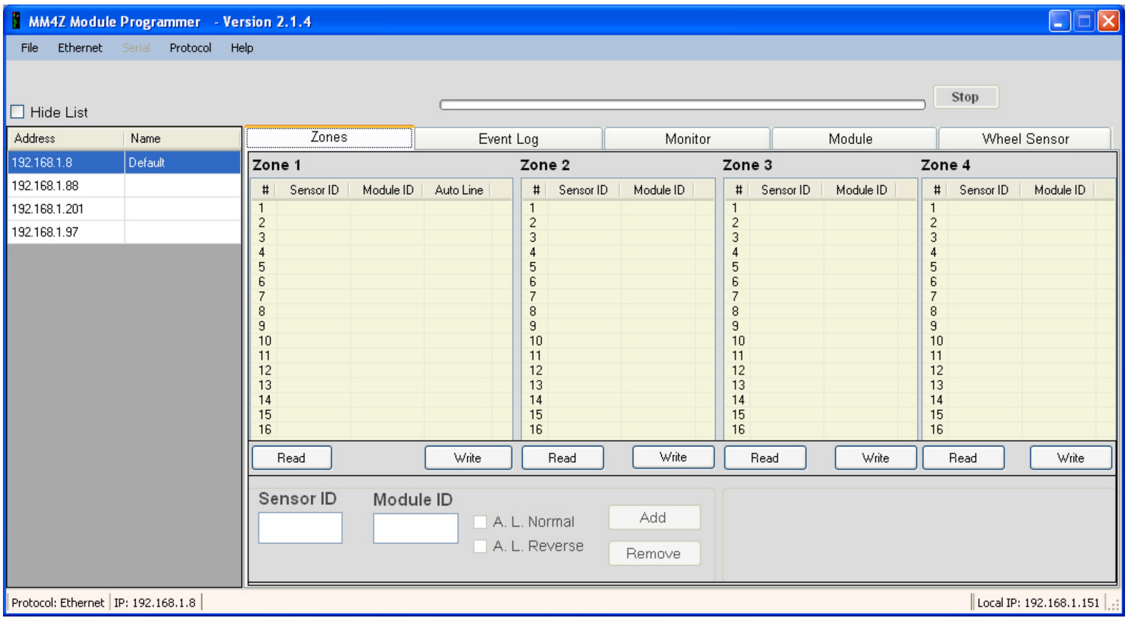


**Figure 7-0 Rail Programmer software**

### 7.1 Zones Tab

Zones Tab is used to group Wheel Sensors into one of the four zones.
**All Wheel Sensor/Module ID numbers are programmed in decimal format.**

**Sensor ID** – The identification number of the Wheel Sensor. Sensor IDs are input in decimal format. Sensor IDs 0, 1, 2, 3 and 153 are reserved and cannot be used.

**Module ID** – The identification number of the MM4Z Module a given Wheel Sensor ID is connected. When programming a sensor that is attached to current Module, Module ID can be left blank or programmed to zero. Module IDs 0 and 255 are reserved and should not be used.

#### 7.1.1 Adding/Removing sensors to a Zone

- **Adding**

Clicking on any of the four Zone lists will enable it for editing.

The “Sensor ID” and “Module ID” textboxes are used to input ID number in decimal format.

The “Add” button is used to store ID number into selected zone lists.

**● Removing**

Selecting a non-blank line from any of the four Zone lists and using the "Remove" button, will remove it from the selected list.

Once information is added or removed from a zone list view, the corresponding "Write" button will change colors to red indicating a change has been made which has not yet been written to the Module.

### 7.1.2 Configuring sensor for "Count Down" direction

To configure a sensor for the count down direction, use a minus sign " – " in front of the Sensor ID number.

For example:

| # | Sensor ID | Module ID | Auto Line |
|---|---|---|---|
| 1 | 04 | | |
| 2 | -05 | | |

**Figure 7-1 Zone List View Example**

Referring to line #1 and line #2 from Figure 7-1, Sensor ID 04 will add its count to the total Zone count, while Sensor ID 05 will subtract its count from the total Zone count.

### 7.1.3 Reading/Writing Zones

While Module is in Run Mode Wait Sequence (See Section 4.2.0), read / write zone operations using Function Code 0x10 are permitted.

While Module is in Run Mode Poll Sequence (See Section 4.2.1), only read operations are permitted.

## 7.2 Event Log Tab

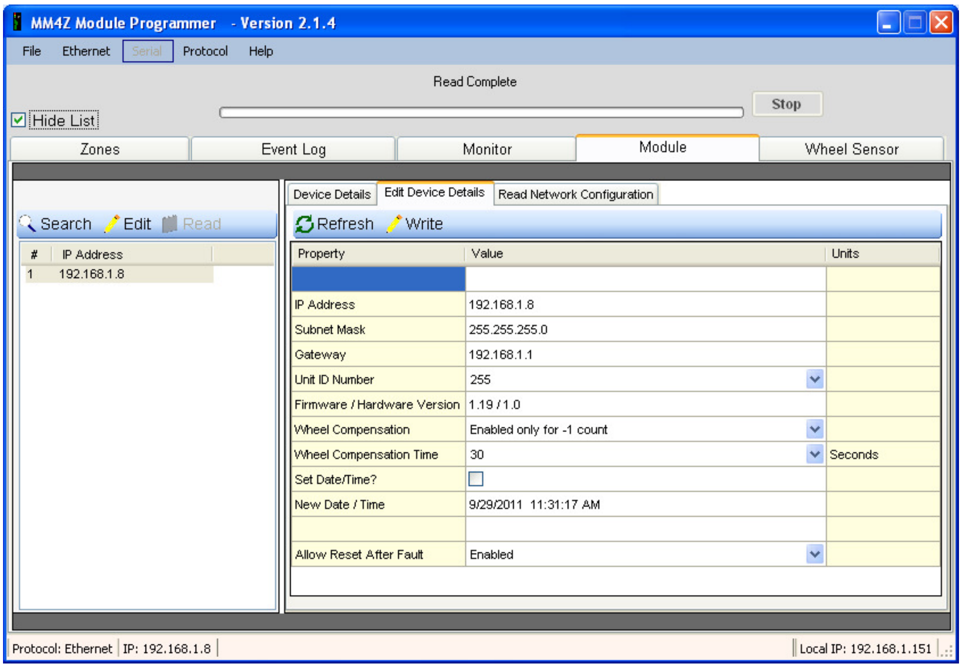
From the event log tab, the full event log can be read from memory and exported to .xls (excel workbook) or tab delimited text file.

## 7.3 Monitor Tab

From the Monitoring Tab, zone counts, current state of inputs/outputs, currently occurring events, and wheel tracking can be monitored in real time.
Note: When monitoring currently occurring events; if more than one event happens at the same time, only one event will be displayed. To display a list of every event, the Event Log tab can be used.

## 7.4 Module Tab

From the Module Tab, Module parameters can be read and edited.



| # | IP Address |
|---|---|
| 1 | 192.168.1.8 |

| Property | Value | Units |
|---|---|---|
| | | |
| IP Address | 192.168.1.8 | |
| Subnet Mask | 255.255.255.0 | |
| Gateway | 192.168.1.1 | |
| Unit ID Number | 255 | |
| Firmware / Hardware Version | 1.19 / 1.0 | |
| Wheel Compensation | Enabled only for -1 count | |
| Wheel Compensation Time | 30 | Seconds |
| Set Date/Time? | □ | |
| New Date / Time | 9/29/2011 11:31:17 AM | |
| | | |
| Allow Reset After Fault | Enabled | |

Protocol: Ethernet | IP: 192.168.1.8 | Local IP: 192.168.1.151

**Figure 7-3 Module list tab**

### 7.4.0 Searching Network for connected Modules

When using the Ethernet option, the search button must first be used to search the network for all Ethernet enabled Modules on the network. When using the Serial option, select the unit ID of the Module you would like to read/edit. If the unit ID of the Module is unknown, the universal unit ID (255) can be used.

**The MM4Z Module ships with default IP Address "192.168.1.8", Default Gateway "192.168.1.1", and 8 host bits**

**The default Network configuration can be reset in Firmware version 1.23 and above using the method outlined in Function 3 of section 1.2.1**

**Searching Network:**

a. The **Search** button will scan your Network for all Ethernet enabled MM4Z Modules.

- **If the Scan Returned NO Results**

  If the scan of your Network returned no results and there is a properly powered up MM4Z Module on your network.

### 7.4.0.0 Verify Ethernet Port Operation

Ensure the MM4Z Module’s Ethernet Port LED1 is illuminated Solid Green or Solid Amber and the LED on your computers Ethernet Network Card is displaying a solid Green light. If no LED lights are illuminated or a Red LED is illuminated on your PCs network card, then no Ethernet link between your PC and the MM4Z module exists.

### 7.4.0.1 Verify the correct Ethernet Network adapter is selected

If your PC has more than one enabled Ethernet adapter, it may be possible that the wrong Ethernet adapter is selected. By default, the PC software will use the first enabled adapter to appear in the PC’s list of network adapters. (See Section 7.6.1 - Network Adapters)

### 7.4.0.2 Verify/Edit the IP Configuration of your PC

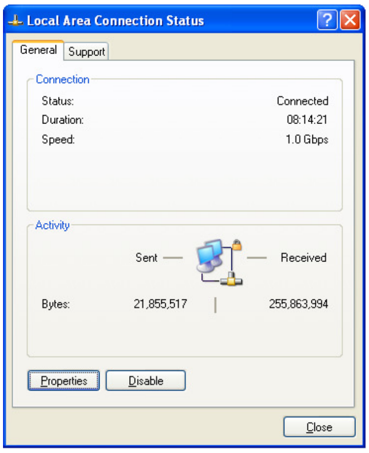
It may be possible that your PC is broadcasting Ethernet communications on a different Gateway or Subnet mask than the MM4Z Module is listening.

This situation may arise when this is the first time configuring the MM4Z Module, or if you are moving a MM4Z Module from one network to another. This can be resolved by temporarily changing the IP configuration of your PC.

**Warning: Record any PC configurations before editing them.**

**To Edit PC Network Configurations:**

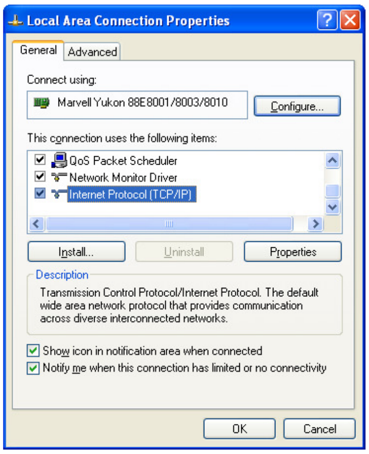
- Open the “Local Area Connection Status”. This can be opened from the Control Panel on your Windows PC or by viewing your network connections.



Status: Connected
Duration: 08:14:21
Speed: 1.0 Gbps
Bytes: 21,855,517 | 255,863,994

**Figure 7- 4 Local Area Connection Status**

- Click the “Properties” button to display the “Local Area Connection Properties”.



Connect using:
Marvell Yukon 88E8001/8003/8010

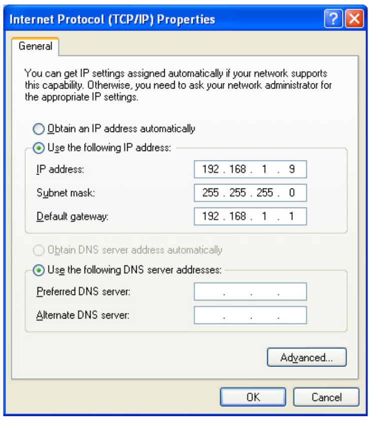
This connection uses the following items:
☑ QoS Packet Scheduler
☑ Network Monitor Driver
☑ Internet Protocol (TCP/IP)

Description
Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.

☑ Show icon in notification area when connected
☑ Notify me when this connection has limited or no connectivity

**Figure 7 - 5 Local Area Connection Properties**

- Scroll down and highlight “Internet Protocol (TCP/IP)”, then click the “Properties” button to display “Internet Protocol (TCP/IP) Properties.



You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically
● Use the following IP address:
IP address: 192 . 168 . 1 . 9
Subnet mask: 255 . 255 . 255 . 0
Default gateway: 192 . 168 . 1 . 1

○ Obtain DNS server address automatically
● Use the following DNS server addresses:
Preferred DNS server: . . .
Alternate DNS server: . . .

**Figure 7- 6 Internet Protocol (TCP/IP) Properties**

- From the "Internet Protocol (TCP/IP) Properties, the IP configuration of the PC can be changed.

In this example, it is assumed that this is the first time changing the IP configuration of the MM4Z Module. The MM4Z Module ships with default IP Address 192.168.1.8. Default Gateway and Subnet mask are left un-configured.

To allow the PC to communicate with this Module, we will change the IP Address of the PC to a unique 192.168.1.x number. We will use IP Address 192.168.1.9.

Once the new IP Address is entered into the IP address field, the Subnet mask should auto fill based on the input IP. The last thing we will do is change the Default Gateway to something that includes the new IP Address, "192.168.1.1" in this example.

After we have changed the IP Address of our computer, the Search can again be used to search for Ethernet enabled MM4Z Modules.

### 7.4.0.3 Restore Default Network Configuration to Module

See Section 1.2.1 Function 3

### 7.4.1 Edit Device Details Tab

From the Edit Device Details Tab, the network configuration, unit ID number, Wheel Compensation function, Wheel Compensation time, Rest functions, and Date/Time can be edited.

### 7.4.2 Read Network Configuration Tab

The Read Network Configuration Tab is used to read Ethernet configuration of the MM4Z Module through a RS485 connection. The Ethernet configuration can only be read immediately after Module power up before any zones are enabled.

## 7.5 Wheel Sensor Tab

From the Wheel Sensor Tab, all Wheel Sensor parameters can be read / written via Ethernet or Serial communication.

### 7.5.0 Port Type

Two port options are available when using the Serial Option.

**a. 4 Pin**

Any of the Zone 1 – Zone 4 connectors are referred to as the 4 Pin connector type.

**b. RJ45**

The RJ45 connector is only available when the Ethernet option is not present. The RJ45 serial connector will be in place of the Ethernet connector.

### 7.5.1 Auto-Detect

Auto-Detect will scan for all connected Wheel Sensors and display them in the “list of Auto-Detected”

### 7.5.2 Scan/Diagnose

Scan/Diagnose function will scan for all connected Wheel Sensors. When a Wheel Sensor is found, its parameters will be read and a test performed before displaying.

### 7.5.3 Read ID

Read ID function will read a connected Wheel Sensor using universal sensor ID 0x99.

### 7.5.4 Send ID

Send ID function will write a new ID number to the desired Wheel Sensor. Send ID function requires a Sensor ID and New Sensor ID number.

### 7.5.5 Read Coil

Read Coil function will read the Threshold, Analog Values, Sensitivity, Baseline, and Count Register of the Wheel Sensor whose ID number is input into the Sensor ID textbox.
If the Monitor checkbox is selected, the Read Coil function will continuously poll Wheel Sensor parameters.

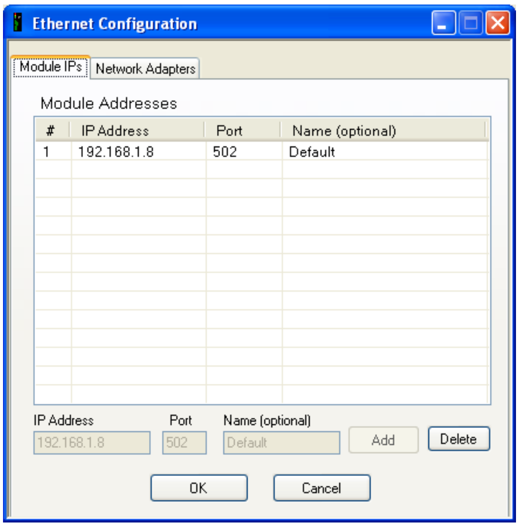
### 7.5.6 Send Coil

Send Coil function will write Threshold values to the Wheel Sensor whose ID number is input into the Sensor ID textbox.

## 7.6 Ethernet Configuration menu

Ethernet Configuration menu is located under the **Ethernet → Communication Setup** menu. Communication Setup menu is used for storing the IP addresses of all the Modules you wish to communicate with.

### 7.6.0 Module IPs Tab



Module Addresses

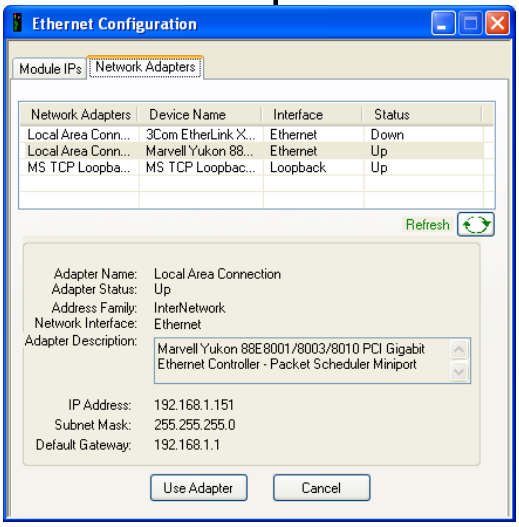
| # | IP Address | Port | Name (optional) |
|---|---|---|---|
| 1 | 192.168.1.8 | 502 | Default |

IP Address: 192.168.1.8
Port: 502
Name (optional): Default

**Figure 7-7 Module IPs Tab**

Clicking on the Module Addresses list will enable the text boxes for inputting the desired IP Addresses. After inputting the IP address and an optional descriptive name, the IP address can be added to the Module Addresses list using the “Add” button.
Note: when using this software to communicate with the Ethernet protocol, the Port number is fixed at 502.

### 7.6.1 Network Adapters Tab



| Network Adapters | Device Name | Interface | Status |
|---|---|---|---|
| Local Area Conn... | 3Com EtherLink X... | Ethernet | Down |
| Local Area Conn... | Marvell Yukon 88... | Ethernet | Up |
| MS TCP Loopba... | MS TCP Loopbac... | Loopback | Up |

Adapter Name: Local Area Connection
Adapter Status: Up
Address Family: InterNetwork
Network Interface: Ethernet
Adapter Description: Marvell Yukon 88E8001/8003/8010 PCI Gigabit Ethernet Controller - Packet Scheduler Miniport

IP Address: 192.168.1.151
Subnet Mask: 255.255.255.0
Default Gateway: 192.168.1.1

**Figure 7-8 Network Adapters Tab**

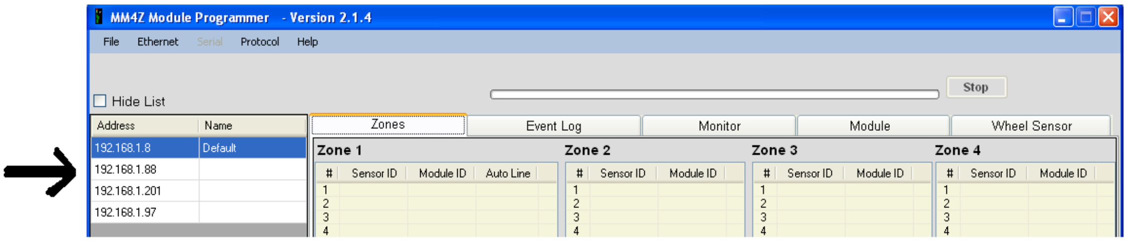
Network Adapters tab will display all installed network adapters on your Windows system.

By default, the MM4Z PC software will select the first enabled Ethernet adapter in the Windows system's list of network adapters.

Selecting any of the enabled network adapters from the list then clicking the "Use Adapter" button, will select that adapter for all further Ethernet communication.

## 7.7 Selecting an IP Address for Communication

After inputting IP addresses using the Module IPs tab (See Section 7.6.0), the input IP addresses will be available in the collapsible IP list on the main program page.



| Address | Name |
|---|---|
| 192.168.1.8 | Default |
| 192.168.1.88 | |
| 192.168.1.201 | |
| 192.168.1.97 | |

**Figure 7-9 Ethernet menu expanded**

## 7.8 Serial menu

From the Serial menu, the COM Port, Port Type, and Upgrade Firmware options can be selected.
The Serial menu is only available when the Serial Protocol is selected.

### 7.8.0 COM Ports

List of all available COM Ports

### 7.8.2 Upgrade Firmware

**To upgrade the MM4Z Module Firmware**

1. Attach MM4Z-06CB cable (See Diagram D.1) to TB4

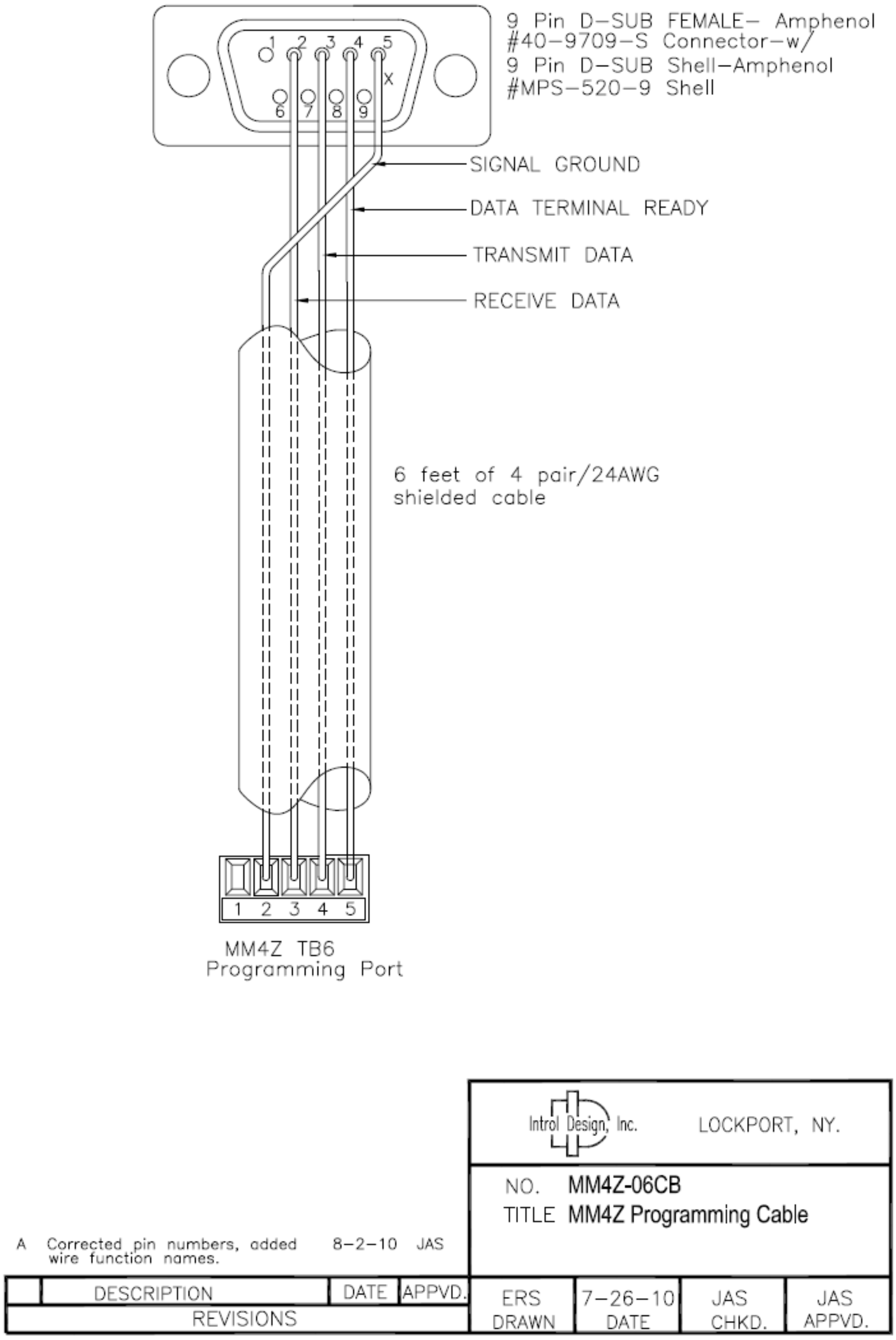
2. Power up Module with Pin 8 of Terminal Block 1 held HIGH to enter Programming Mode (See Section 4.2)

3. Use the **Serial** → **COM Port** menu of the MM4Z PC software to select COM Port MM4Z-06CB cable is connected to.

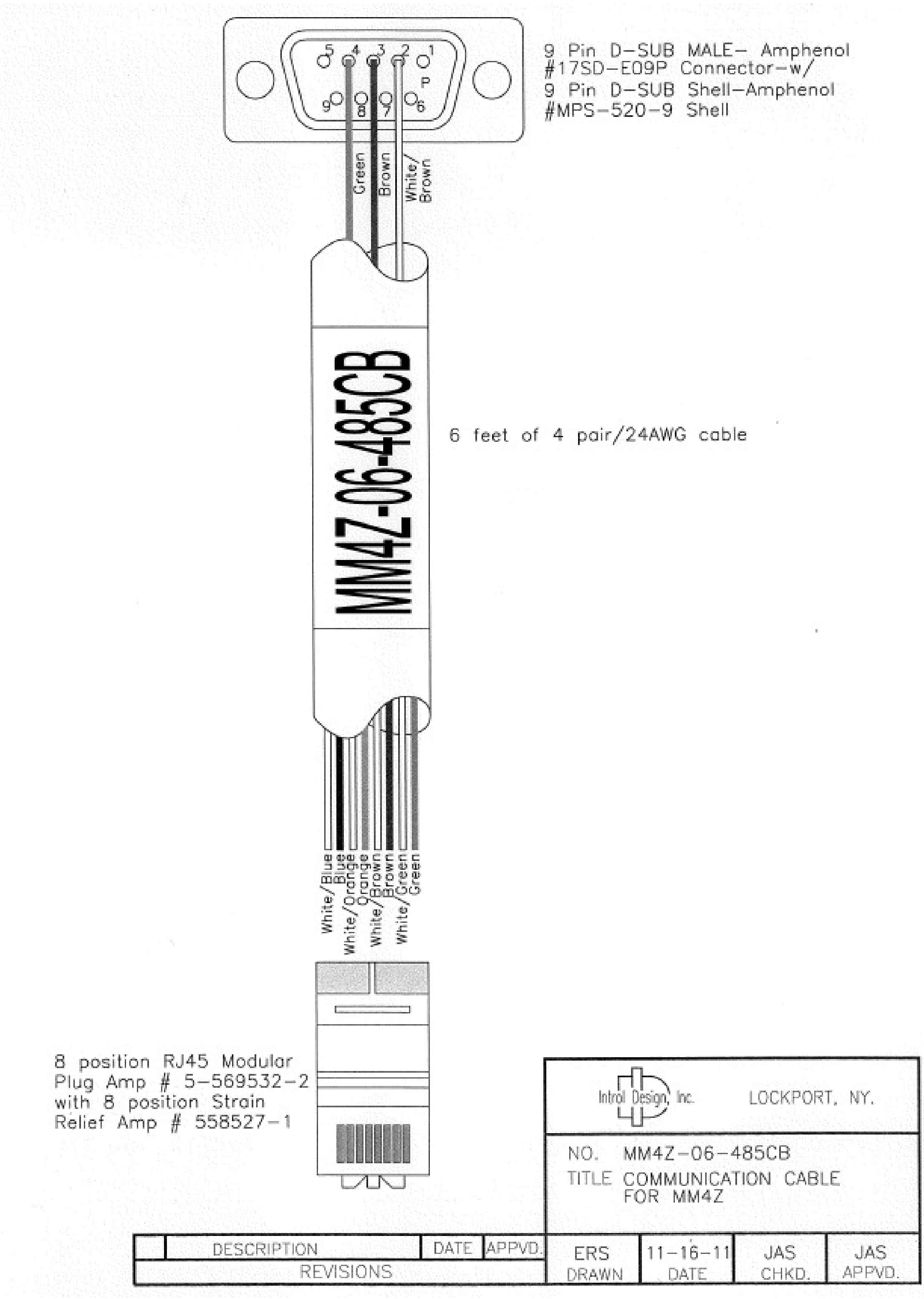
4. Upgrade Firmware: Click **Serial** → **Upgrade Firmware.** Navigate to the *.ifrm upgrade file. The Module upgrade will begin immediately.

5. Once Firmware Upgrade completes successfully, Detach Programming cable and Power up module with Pin 8 of Terminal Block 1 held LOW or left floating to enter Run Mode (See Section 4.2).

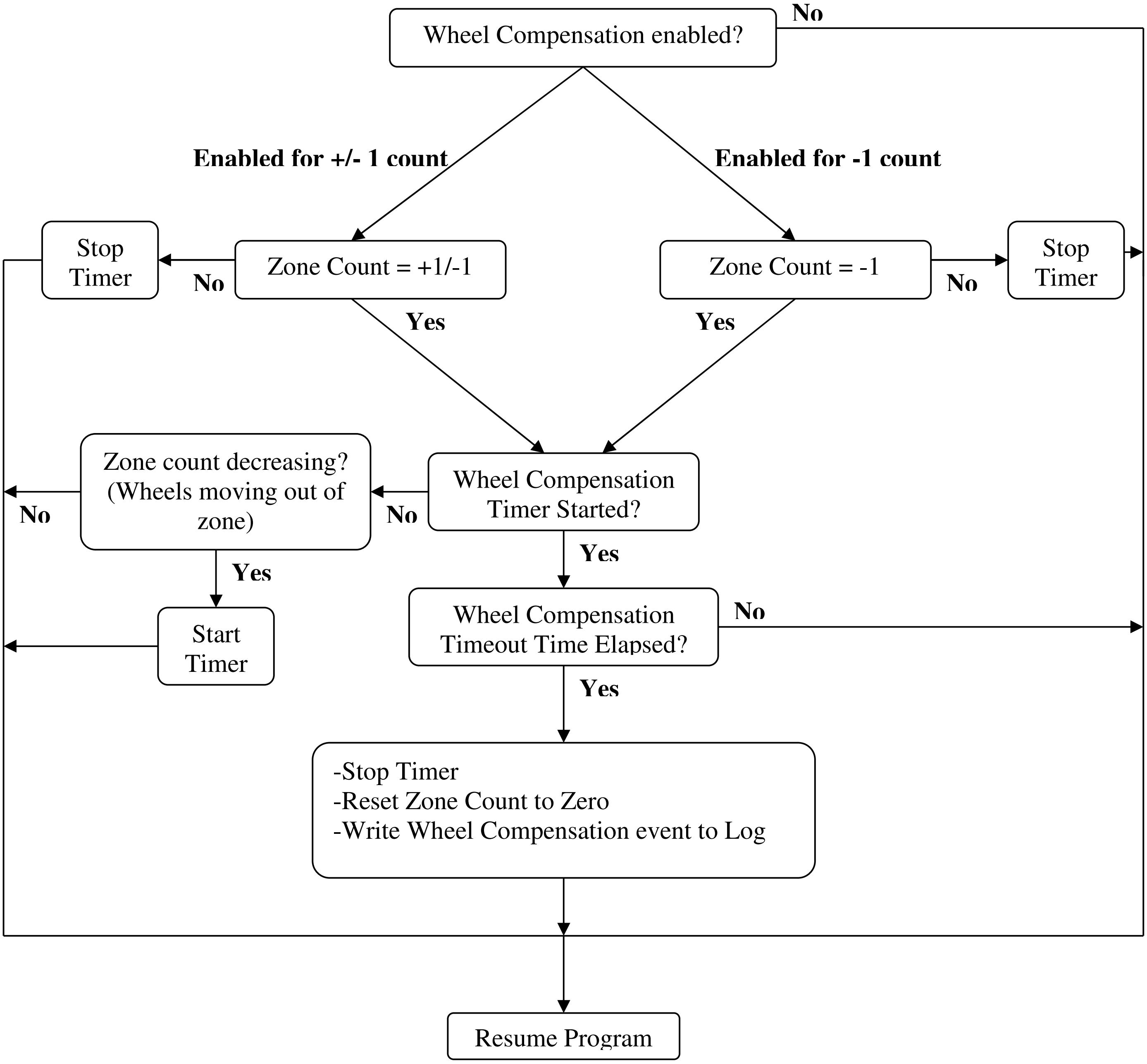
# Diagrams


1 2 3 4 5
x
6 7 8 9
9 Pin D-SUB FEMALE- Amphenol #40-9709-S Connector-w/
9 Pin D-SUB Shell-Amphenol #MPS-520-9 Shell
SIGNAL GROUND
DATA TERMINAL READY
TRANSMIT DATA
RECEIVE DATA
6 feet of 4 pair/24AWG shielded cable
1 2 3 4 5
MM4Z TB6
Programming Port
Introl Design, Inc.
LOCKPORT, NY.
NO. MM4Z-06CB
TITLE MM4Z Programming Cable
A Corrected pin numbers, added wire function names. 8-2-10 JAS
DESCRIPTION
DATE
APPVD.
REVISIONS
ERS
DRAWN
7-26-10
DATE
JAS
CHKD.
JAS
APPVD.

## D.1 MM4Z-06CB Programming Cable



## D.2 RS485 Communication Cable


Wheel Compensation enabled?
No
Enabled for +/- 1 count
Enabled for -1 count
Stop Timer
Zone Count = +1/-1
No
Zone Count = -1
No
Stop Timer
Yes
Yes
Zone count decreasing? (Wheels moving out of zone)
No
No
Wheel Compensation Timer Started?
Yes
Yes
Start Timer
Wheel Compensation Timeout Time Elapsed?
No
Yes
-Stop Timer
-Reset Zone Count to Zero
-Write Wheel Compensation event to Log
Resume Program

## D.3 Wheel Compensation Flow Diagram

# ANNEX A. Revision History

### A.1 Revision 1.0

**Initial Revision**

### A.2 Revision 1.1

**Updated procedure to change Ethernet configurations using the MM4Z Programmer software**

### A.3 Revision 1.2

**Updated MM4Z-06CB Programming Cable Drawing**

### A.4 Revision 1.2.1

**Update Event Code 0x0010, 0x0014, and 0x0015**

### A.5 Revision 1.2.2

**Updated photos for new PC software**
**General reformatting**

### A.6 Revision 1.3

**Expanded explanation of events**
**Added Wheel Compensation flow diagram**
**RS485 Communication cable diagram**

### A.7 Revision 1.4

**Definition/example for Function Code 0x05 and 0x06**

### A.8 Revision 1.5

**Definitions for Function Code 0x11**