Overview

The MTM Power Module (MTM-PM-1 or S65-MTM-PM-1), as part of Acroname® MTM (Manufacturing Test Module) product series, is a software-controlled voltage and current limiting modular power supply, designed for MTM-based manufacturing or R&D test systems. The MTM-PM-1 allows MTM system designers to easily and modularly add power regulation and control to their test system designs.

The MTM-PM-1 provides stable, consistent and robust power to a wide range of devices and is optimized for devices using LiPo or similar batteries. Switch mode and linear mode are available, giving MTM-PM-1's ultra-fast transient response and making it ideal for use in RF noise-sensitive applications (GSM, UMTS, LTE, CDMA, etc.).

Built using Acroname's industry-proven and well-adopted BrainStem® technology, resources on the MTM-PM-1 are controlled via Acroname's powerful and extensible BrainStem technology and software APIs.

Typical Application

- Manufacturing functional testing
- Validation testing
- Automated test development
- Embedded system development
- Battery emulation

System Features

- Rail0
  - Software-controllable, regulated output with ~1mV resolution, up to a 5V/3A current limit
  - Switched and linear regulation mode stages
  - Kelvin sense
  - Rail output current voltage-mirror
  - Rail enabled status indicator
- Rail1
  - Unregulated pass-through rail from V_{supply} (software enabled and current limiting)
- 2 Digital GPIOs (overvoltage and current protected)
- 1 BrainStem I²C FM+ (1Mbit/s) bus
- Temperature sense and overtemperature protection
- Optional K-Type thermocouple input for remote sensing (ordering option)

Description

As part of Acroname's MTM series, the MTM-PM-1 is a key component for manufacturing test and R&D of devices requiring highly accurate, controllable power supply and measurement in an MTM-based test system. BrainStem interface and APIs are at https://acroname.com/reference.

The MTM-PM-1 implements an onboard BrainStem controller running an RTOS (Real-Time Operating System), which provides a host connection, independent operating capability and the BrainStem interface, for control of the MTM resources identified in this datasheet (Rail0, Rail1, GPIO, etc.). These resources can be controlled from a host computer over USB or in a network of MTM modules.

The MTM-PM-1 provides two main power outputs. Rail0 is a fully regulated output that includes a switch-mode stage as well as a linear stage. Rail1 is an unregulated, switchable output that passes through power directly from V_{supply}. In noise-sensitive applications, the MTM-PM-1 can be operated in full linear power supply mode; no potential RF noise from switchers. When thermal and efficiency operation is more important, utilize the first-stage, pre-conditioning switch mode power supply (SMPS) followed by the high-accuracy, ultra-fast response linear stage.

Within the MTM platform architecture, the MTM-PM-1 module can operate either independently or as a component in a larger network of MTM modules. Each MTM-PM-1 is uniquely addressable and controllable from a host by connecting via the onboard USB connection, the card-edge USB input or through other MTM modules on the local BrainStem bus.

Acroname's BrainStem link is established over the selected input connection. The BrainStem link allows a connection to the onboard controller and access to the available resources in the MTM-PM-1. The MTM-PM-1 can then be controlled via a host running BrainStem APIs or it can operate independently by running locally embedded, user-defined programs based on Acroname's BrainStem Reflex language in the RTOS.

IMPORTANT NOTE

The MTM-PM-1 utilizes a PCIe connector interface but is for use strictly in MTM-based systems. It should never be installed in a PCI slot of a host computer directly. Insertion into a PC or non-MTM system could cause damage to the PC.

Absolute Maximum Ratings

Stresses beyond those listed under ABSOLUTE MAXIMUM RATINGS can cause permanent damage to the device. These are stress ratings only and functional operation of the device at these or any other conditions beyond those indicated under RECOMMENDED OPERATING CONDITIONS is not implied. Exposure to absolute-maximum rated conditions for extended periods affects device reliability and may permanently damage the device.

Voltage Rating	Minimum	Maximum	Units
Input Voltage, Vsupply	-13.2	13.2	V
I2C0 SDA, SCL	-0.5	13.2	V
UART TX/RX	-0.5	13.2	V
DIO0-1	-0.5	13.2	V
RAIL0 Kelvin V+	-0.5	13.2	V
Module Address 0-3	-0.5	13.2	V
Reset	-0.5	13.2	V
RAIL0 Enable Status	-0.5	13.2	V
RAIL0 Current Mirror	-0.5	13.2	V
USB D+, D-	-0.5	5.5	V
USB Vbus	-0.5	6.0	V
RAIL0 V+	-0.5	13.2	V
RAIL1 V+	-0.5	Vsupply	V
Thermocouple +/-	-10	+25	V

Table 1: Absolute Maximum Voltage Ratings

Current Rating	Minimum	Maximum	Units
Input Current, I_{supply}	0.0	5.5	A

Table 2: Absolute Maximum Current Ratings

Temperature Rating	Minimum	Maximum	Units
Rail0 Thermal Shutdown	-	125	°C

Table 3: Absolute Maximum Temperature Ratings

The MTM system is designed to be used in a system where V_{supply} is the highest voltage connected to all MTM modules. Each module is designed to withstand V_{supply} continuously connected to all IOs, excepting those specified above, including accidental reverse polarity connection between V_{supply} and ground (0V). As with all products, care should be taken to properly match interface voltages and ensure a well-architected current-return path to ground. As with all devices utilizing USB interfaces, care should be taken to avoid ground loops within the USB subsystem. When using the USB interface, ground must be at 0V potential to avoid damaging connected host systems.

Handling Ratings

Parameter	Conditions/Notes	Minimum	Typical	Maximum	Units
Ambient Operating Temperature, T_A	Non-Condensing	0	25	70	°C
Relative Humidity Range	Non-Condensing	5	-	95	%RH
Storage Temperature, T_{STG}		-10	-	85	°C
Electrostatic Discharge, V_{ESD}	IEC 61000-4-2, level 4, contact discharge to edge connector interface	-8	-	+8	kV

Table 4: Handling Ratings

Recommended Operating Ratings

Specifications are valid at 25°C unless otherwise noted. Intended for indoor use only.

Parameter	Conditions/Notes	Minimum	Typical	Maximum	Units
Input Voltage, V_{supply}		6.0	-	12.0	V
Voltage to any IO pin		0	-	3.3	V
Voltage to any I2C pin		0	-	3.3	V
Relative Humidity Range	Non-Condensing	5	-	95	%RH

Table 5: Recommended Operating Ratings

Block Diagram

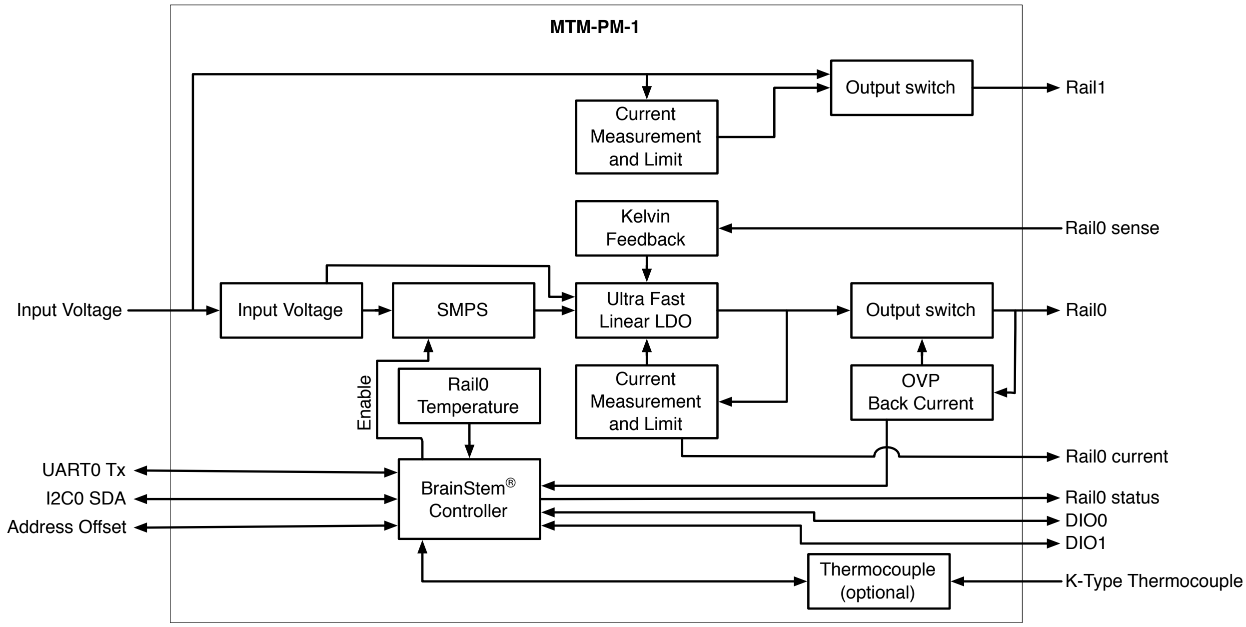


Figure 1: MTM-PM-1 Block Diagram

Typical Performance Characteristics

Specifications are valid at 25°C unless otherwise noted. Indoor application use only. Sample rates are typically limited by the USB throughput of the host operating system except where bulk capture is supported.

Parameter	Conditions/Notes	Minimum	Typical	Maximum	Units
Base Current Consumption, I_{supply}	V_{supply}=6V V_{supply}=12V	- -	107 122	- -	mA
Rail0 Output Voltage, V_{RAIL0}	Software controlled	1.8	-	5.0	V
Rail0 Output Current Limit	Software controlled	0.001	-	3.0	A
Rail0 Turn-on Time		-	-	1.0	s
Rail1 Output Current Limit	Software controlled	0.001	-	3.0	A
Digital Input Logic High, V_{IH}		2.15	-	-	V
Digital Input Logic Low, V_{IL}		-	-	1.1	V
Digital Input Leakage Current	Mode set Input or High-Z	-	110	-	µA
Digital Input Resistance	Mode set Input or High-Z	-	4.25	4.45	MΩ
Digital Output Logic High, V_{OH}		-	3.3	-	V
Digital Output Drive Current	Output high; short to GND V_{output} = 0.9* V_{OH}	- -	20.0 3.15	30.0 -	mA
Digital Output Sink Current	Output low; short to V_{supply}	-	-20.0	-30.0	mA
Digital Output Short Duration	Output high	-	Infinite	-	hours
Digital Output Overvoltage Duration	V_{supply} on pin	-	Infinite	-	hours
Digital Sample Rate[1]	via USB link, C++ Reflex	- -	1000 8200	- -	Hz
Reset Low Threshold		-	1.2	-	V
I2C SDA, SCL Pins		0.0	3.3	-	V
Rail0 Thermal Shutdown		-	-	125	°C
Rail0 Thermal Shutdown Hysteresis	After thermal shutdown event	100.0	-	-	°C
Rail0 Output Voltage Ripple	Linear Mode	-	-	21	mV
Rail0 Output Voltage Ripple	Switcher Mode	-	-	31	mV
Rail0 Voltage Regulation Accuracy	150mA load	-	0.5	-	%
Rail0 Voltage Readback Accuracy		-	0.5	-	%
Rail0 Voltage Output Ripple	150mA load	-	13	20	mV
Rail0 Voltage Control Resolution		-	1.5	-	mV
Rail0 Linear-mode Drop Out		-	0.8	1.2	V
Rail0 Switcher-mode Supply	V_{supply}	7.0	-	-	V
Rail0 Thermal Drift	Full temperature range	-	-	1.5	mV
Rail0 Current Voltage-Mirror[2]			1.0		V/A
Rail0 Current Mirror Zero-offset		-	15	20	mV
Rail0 Current Mirror Output Accuracy	$I_{out} \leq$ 500mA I_{out} > 500mA	- -	2 0.5	5 1	%

[1] Host dependent, test was done as a single instruction, subsequent instructions may affect performance.
[2] Current output voltage-mirror must be connected to high impedance.

Rail0 Transient Load Response	1A step load at 5µs[3] 1A step load at 28µs[1] 2A step load at 28µs[1] 3A step load at 28µs[1]	- - - -	80 10 10 25	100 20 25 35	mV
Rail0 Enable Time[4], 0µF [5]	V_{rail} = 1.8V, limit=3A V_{rail} = 3.3V, limit=3A V_{rail} = 3.8V, limit=3A V_{rail} = 5.0V, limit=3A	- - - -	80 125 140 175	- - - -	µs
Rail0 Enable Time[2], 4.7uF [3]	V_{rail} = 1.8V, limit=3A V_{rail} = 3.3V, limit=3A V_{rail} = 3.8V, limit=3A V_{rail} = 5.0V, limit=3A	- - - -	110 165 175 230	- - - -	µs
Rail0 operational mode auto Input voltage thresholds	SMPS disable min voltage SMPS enable max voltage	- -	7.2 6.8	- -	V
Rail1 On resistance (R_{on})		-	70	-	mΩ
Rail1 Enable Time, 0 uF [3]		-	570	-	µs
Rail1 Enable Time, 4.7 uF [3]		-	590	-	µs
Current Limit Accuracy	Limit <250mA Limit ≥250mA	- -	25 10	- -	mA %

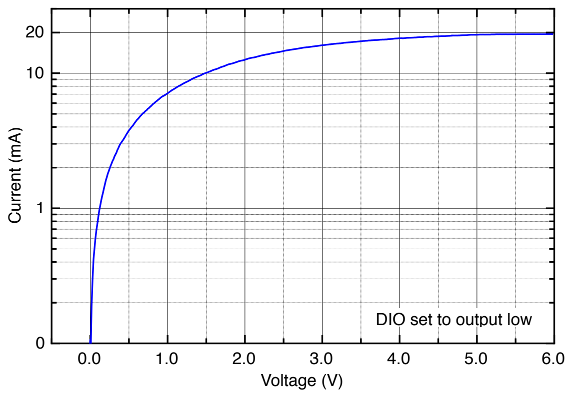
Table 6: Typical Performance Characteristics

[3] Kelvin sense connected and enabled; 5µs is maximum voltage drop due to step load; 28µs for 3GPP reference.

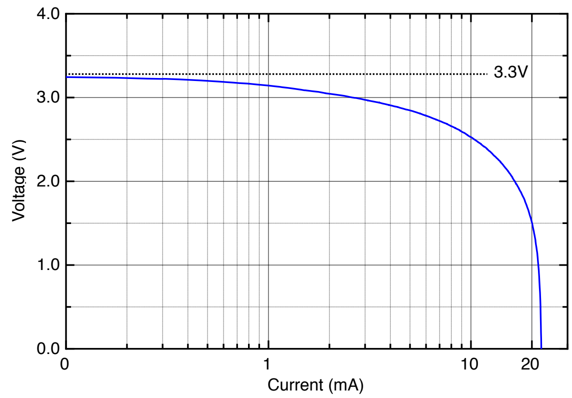
[4] Rise time to within 90% of V_{rail}.

[5] Enable time depends heavily on the operational load, so the maximum rise time will be application-dependent.

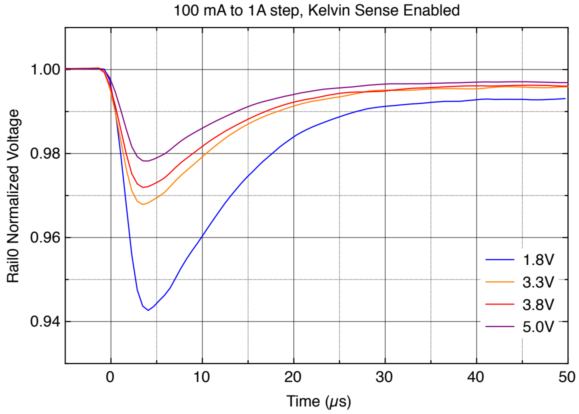
DIO Output Sink Current vs. Applied Voltage



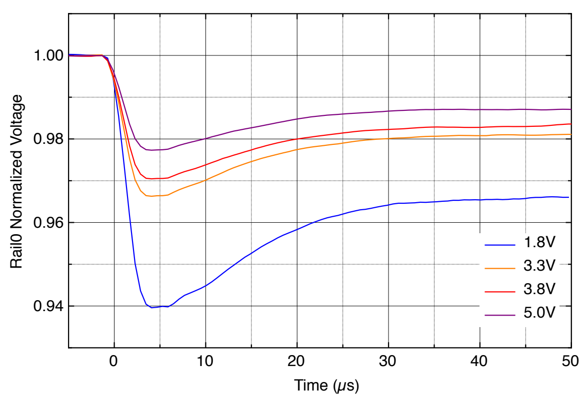
DIO Output High Voltage vs. Current Load

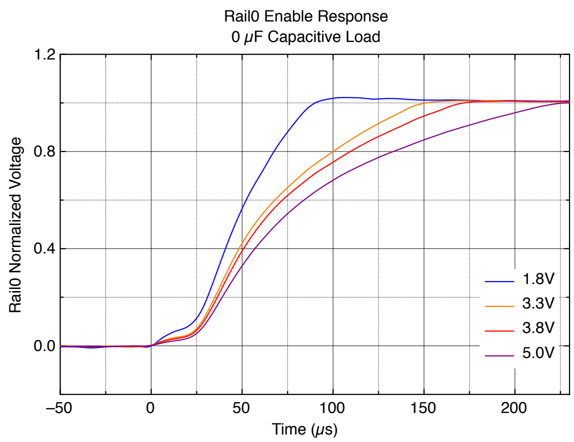


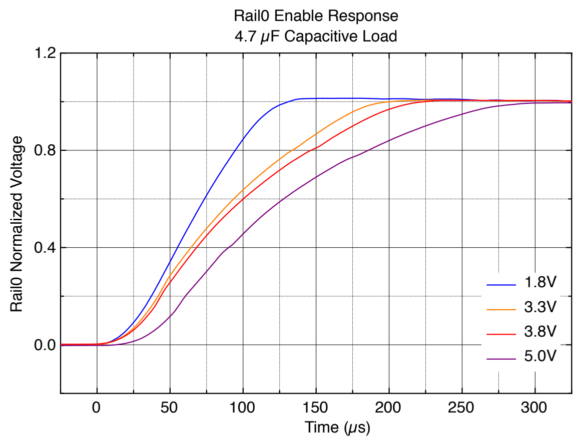
Load Transient Response
100 mA to 1A step, Kelvin Sense Enabled

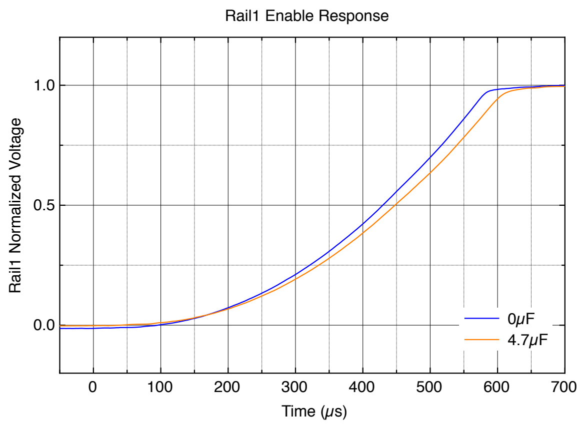


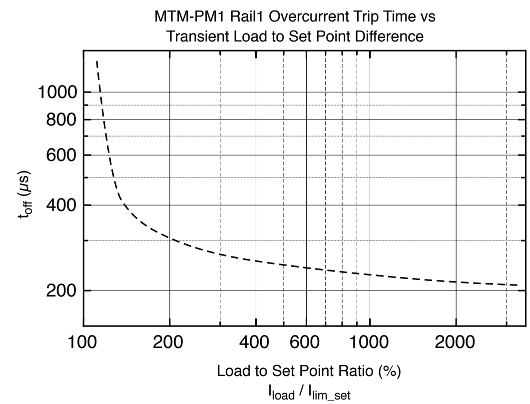
Load Transient Response
100mA to 1A step, Kelvin Sense Disabled

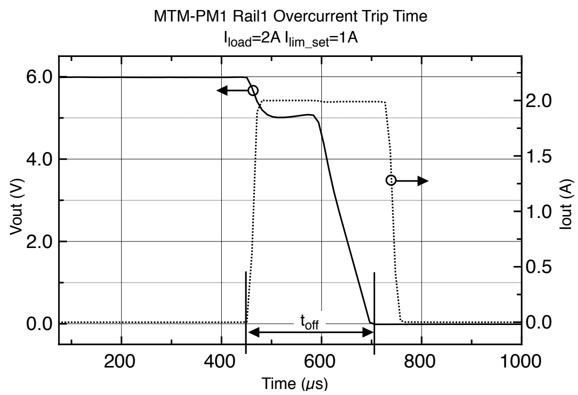



Rail0 Enable Response
0 μF Capacitive Load
Rail0 Normalized Voltage
Time (μs)
1.8V
3.3V
3.8V
5.0V


Rail0 Enable Response
4.7 μF Capacitive Load
Rail0 Normalized Voltage
Time (μs)
1.8V
3.3V
3.8V
5.0V


Rail1 Enable Response
Rail1 Normalized Voltage
Time (μs)
0μF
4.7μF

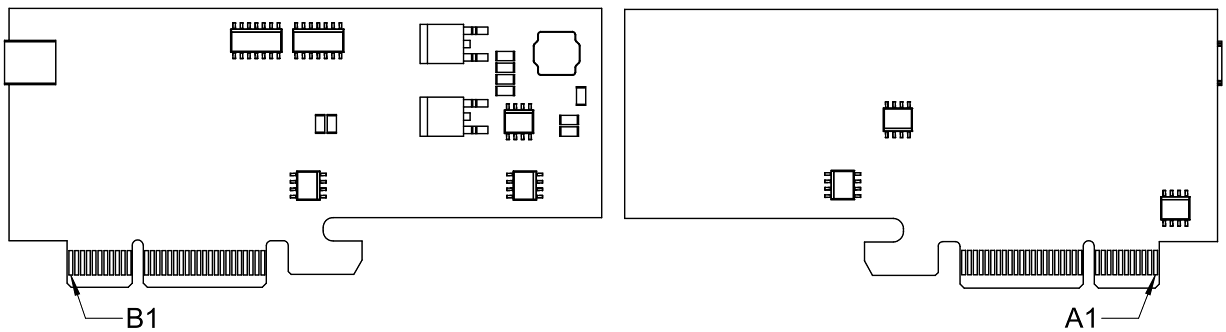

MTM-PM1 Rail1 Overcurrent Trip Time vs
Transient Load to Set Point Difference
t_off (μs)
Load to Set Point Ratio (%)
I_load / I_lim_set


MTM-PM1 Rail1 Overcurrent Trip Time
I_load=2A I_lim_set=1A
Vout (V)
Iout (A)
t_off
Time (μs)

Pinout Descriptions

WARNING: MTM modules use a PCIe connector interface that is common in most desktop computers; however, they are NOT intended nor designed to work in these devices. Do NOT insert this product into any PCIe slot that wasn't specifically designed for MTM modules, such as a host PC. Installing this module into a standard PCI slot will result in damage to the module and the PC.

The MTM edge connector pin assignments are shown in the following table. Please refer to Table 5: Recommended Operating Ratings for appropriate signal levels.



Pins Common to all MTM Modules

Edge Connector Side A	Edge Connector Side A Description	Edge Connector Side B	Edge Connector Side B Description
1	GND	1	Input Voltage, V_{supply}
2	GND	2	Input Voltage, V_{supply}
3	GND	3	Input Voltage, V_{supply}
4	GND	4	Input Voltage, V_{supply}
5	Reset	5	Input Voltage, V_{supply}
6	GND	6	Reserved, Do Not Connect
7	GND	7	Reserved, Do Not Connect
8	I^2C SCL	8	GND
9	I^2C SDA	9	GND
10	GND	10	UART0 Transmit
11	GND	11	UART0 Receive
12	Module Address Offset 0	12	Module Address Offset 2
13	Module Address Offset 1	13	Module Address Offset 3

Table 7: Pins Common to all MTM Modules

Pins Specific to MTM-PM-1

Edge Connector Side A	Edge Connector Side A Description	Edge Connector Side B	Edge Connector Side B Description
14	Reserved, Do Not Connect	14	USB Upstream Data +
15	K-Type Thermocouple +	15	USB Upstream Data -
16	K-Type Thermocouple -	16	Reserved, Do Not Connect
17	Reserved, Do Not Connect	17	Reserved, Do Not Connect
18	RAIL0 Kelvin Sense Negative Return[6]	18	RAIL0 Kelvin Sense Positive Return
19	GND	19	RAIL0 +
20	GND	20	RAIL0 +
21	GND	21	RAIL0 +
22	GND	22	RAIL0 +
23	Digital IO 1	23	Digital IO 0
24	RAIL0 Current Mirror (1V/1A)	24	RAIL0 Power Enable Status
25	Reserved, Do Not Connect	25	RAIL1 +
26	Reserved, Do Not Connect	26	RAIL1 +
27	Reserved, Do Not Connect	27	RAIL1 +
28	Reserved, Do Not Connect	28	RAIL1 +
29	Reserved, Do Not Connect	29	Reserved, Do Not Connect
30	Reserved, Do Not Connect	30	Reserved, Do Not Connect
31	Reserved, Do Not Connect	31	Reserved, Do Not Connect
32	Reserved, Do Not Connect	32	Reserved, Do Not Connect

Table 8: Pins Specific to MTM-PM-1

[6] RAIL0 Kelvin Sense Negative return is electrically connected to GND.

Module Hardware and Software Default Values

Software Control

The MTM-PM-1 module firmware is built on Acroname's BrainStem technology and utilizes a subset of BrainStem entity implementations that are specific to the hardware's capabilities. Table 9 details the BrainStem API entities and macros used to interface with the MTM-PM-1 module. For C and C++ developers, these macros are defined in `aMTMPM1.h` from the BrainStem development package. For Python development, the module `MTMPM1` class defines the extent of each entity array.

While Table 9 lists the BrainStem API entities available for this module, not all entity methods are supported by the MTM-PM-1. For a complete list of supported entity methods, see Table 17. Note that available method options may vary by entity index, as well as by entity, and calling an unsupported entity option will return an appropriate error (e.g.: `aErrInvalidEntity`, `aErrInvalidOption`, `aErrMode`, or `aErrUnimplemented`) as defined in `aError.h` for C and C++ and the `Result` class in Python.

All API example code snippets that follow are pseudocode loosely based on the C++ method calls - Python and Reflex are similar. Please consult the BrainStem Reference for specific implementation details[8].

Parameter	Index	Macro Name or Implemented Options	Notes
Module Definitions:			
Module Base Address	6	aMTMPM1_MODULE_BASE_ADDRESS	See aMTMPM1.h
Entity Class Definitions:			
digital Entity Quantity	2	aMTMPM1_NUM_DIGITALS	
rail Entity Quantity	2	aMTMPM1_NUM_RAILS	
Adjustable (`rail[0]`)	0	aMTMPM1_RAIL0	
Vsupply pass-through (`rail[1]`)	1	aMTMPM1_RAIL1	
i2c Entity Quantity	1	aMTMPM1_NUM_I2C	
Store Entity Quantity	2	aMTMPM1_NUM_STORES	
system Entity Quantity	1		
timer Entity Quantity	8	aMTMPM1_NUM_TIMERS	
`temperature` Entity Quantity	1	aMTMPM1_NUM_TEMPERATURES	
`app` Entity Quantity	4	aMTMPM1_NUM_APPS	
`pointer` Entity Quantity	4	aMTMPM1_NUM_POINTERS	

Table 9: MTM-PM-1 Hardware and Software Default Values[7]

[7] Refer to `aMTMPM1.h` within the BrainStem Development Kit download for actual file.

Capabilities and Interfaces

BrainStem Link and Module Networking

A BrainStem link can be established that will give the user access to the resources available on the MTM-PM-1. The module can then be controlled via a host running BrainStem APIs or operated independently by running locally embedded, user-defined programs based on Acroname's BrainStem Reflex language in the RTOS.

A BrainStem link to the MTM-PM-1can be established via one of three (3) interfaces: the onboard USB connection, the card-edge USB connection, or through another MTM module using the BrainStem protocol (more on this interface below). For the USB connection options, once the MTM-PM-1 is attached to a host machine, a user can connect to it via software API:

```
stem.link.discoverAndConnect(linkType,
serialNumber, modelNumber)
```

The MTM-PM-1 can also work within a network of other Brainstem modules, such as in a test fixture, to give access to the capabilities of all networked modules. On the MTM platform, networked modules communicate using the Brainstem protocol, which is transmitted over I^2C. Each MTM-PM-1is uniquely addressable via hardware or software to avoid communication conflicts on the I^2C bus. A software offset can be applied as follows:

```
stem.system.setModuleSoftwareOffset(address)
```

Upstream USB Connectivity Options

The MTM-PM-1 supports upstream USB connections (to communicate to a host PC) via the mini-B connector, or through pins B14 and B15 of the PCIe edge connector. The module defaults to using the edge connector and will switch to the mini-B connector if 5V is present on V_{bus} at the mini-B connector.

Module Address Hardware Offset Configuration

A hardware offset is one of two ways to modify the module's address on the BrainStem network. Using hardware offset pins is useful when more than one of the same type of module is installed on a single BrainStem network. Applying a different hardware offset to each module of the same type in one network allows for all the modules to seamlessly and automatically configure the network for inter-module communication. Further, modules can be simply swapped in and out of the network without needing to pre-configure a module's address before being added to a network. Finally, when a system has more than one of the same type of module in a network, the module address hardware offset can be used to determine the module's physical location and thus its interconnection and intended function. For detailed information on BrainStem networking see the BrainStem Reference[8].

Each hardware offset pin can be left floating or pulled to ground with a 1kΩ resistor or smaller (pin may be directly shorted to ground). Pin states are only read when the module boots, either from a power cycle, hardware reset, or software reset. The hardware offset pins are treated as a binary number which is multiplied by 2 and added the to the module's base address. The hardware offset calculation is detailed in the following table.

HW Offset Pin				Address Offset	Module Base Address	Final Module Address
3	2	1	0			
NC	NC	NC	NC	0	6	6
NC	NC	NC	0	2	6	8
NC	NC	0	NC	4	6	10
NC	0	NC	NC	8	6	14
0	NC	NC	NC	16	6	22
0	NC	NC	0	2+16	6	24

Table 10: Module Address Hardware Offset Examples

System Entities

Every BrainStem module includes a single System Entity. The System Entity allows access to configuration settings such as the module address and I^2C rate, measurements such as input voltage, control over the user LED, and many more.

Saving Entity Settings

Some entities can be configured and saved to non-volatile memory. This allows a user to modify the MTM-PM-1's startup and operational behavior *away* from the factory default settings. Saving system settings creates a new default and often requires a reboot of the MTM-PM1 to take effect; see Table 11: Entities Values Saved by system.save() for relevant settings. Use the following command to save changes to system settings before reboot::

```
stem.system.save()
```

Saved Configurations	
Software Offset	I2C Rate
Router Address	I2C Pullup State
Heartbeat Rate	Boot Slot

Table 11: Entities Values Saved by system.save()

Store Entities

Every BrainStem module includes several Store entities and onboard memory slots to load Reflex files (for details on Reflex, see BrainStem Reference[8]). One Reflex file can be stored per slot. Store[0] refers to the internal flash memory, with 12 available slots, and store[1] refers to RAM, with 1 available slot.

Rail Entities

Rails on the MTM-PM-1 module are powerful (no pun intended); they allow other devices and peripherals to consume power from the MTM-PM-1 module in a precisely controlled fashion. Two (2) different rails are available for use: a software-adjustable voltage rail (`rail0`), and input voltage pass-through rail (`rail1`). These rails are accessed through an array of BrainStem rail class entities. The MTM-PM-1 module implements a subset of the BrainStem rail class for each of these rails. Table 17: Supported MTM-PM-1 BrainStem Entity API Methods summarizes the implemented rail entity options for each entity index.

Enabling Rails

Both rails can be switched on or off using the `setEnable` API.

```
stem.rail[0].setEnable(state)
```

RAIL0 Voltage Setting

The `rail0` voltage setpoint is adjustable through the API:

```
stem.rail[0].setVoltageSetpoint(microvolts)
stem.rail[0].getVoltageSetpoint(microvolts)
```

Setting values outside the allowable range will return an error.

RAIL0 and RAIL1 Measurements

Measured rail voltages and currents can be read from the API:

```
stem.rail[0].getVoltage(microvolts)
stem.rail[0].getCurrent(microamps)
```

RAIL0 Operational Mode

`rail0` contains an optional SMPS (switch-mode power supply) pre-regulation stage power path. The power path can be configured through the API:

```
stem.rail[0].getOperationalMode(mode)
stem.rail[0].setOperationalMode(mode)
```

The `mode` parameter is an integer that corresponds to one of the following valid operational modes:

Mode	Rail[0] Operational Mode Description
0	railOperationalModeAuto (default)
1	railOperationalModeLinear
3	railOperationalModeSwitcherLinear

Table 12: Rail[0] Operational Modes

The default mode railOperationalModeAuto automatically enables or disables the SMPS depending on the module input voltage.

For applications such as RF receiver sensitivity testing, it is suggested to operate only in linear regulation mode to eliminate potential EMI sources. When operating in linear mode, make sure to consider power dissipation through the linear regulation stage; a higher input voltage will result in higher power dissipation. When linear mode is desired as well as high current operation, it is recommended to run the input voltage close to the MTM-PM-1 module's minimum input voltage to reduce power regulation losses. Switch-mode power supply operation will allow a broader range of input voltages while maintaining high current demand limits.

RAIL0 and RAIL1 Operational State

Use the API to determine the active operational state of each rail:

```
stem.rail[0].getOperationalState(state)
```

The value *state* is a 32-bit value, composed of several fields as summarized in Table 13. Note that the relevant fields differ between `rail0` and `rail1`.

Bits	RAIL Operational State Field Description	Rails
0	Initializing (railOperationalState_Initializing)	`rail0` `rail1`
1	Enabled (railOperationalState_Enabled)	
2	Fault (railOperationalState_Fault)	
3-15	Reserved	
8-15	Hardware Configuration (railOperationalState_HardwareConfiguration)	`rail0`
16-17	Reserved	
18	Overcurrent Fault "OC" (railOperationalStateOverCurrentFault)	`rail1`
19-20	Reserved	
21	Overtemperature Fault "OT" (railOperationalStateOverTemperatureFault)	`rail0`
22-31	Reserved	

Table 13: Rail Operational Sate

The Initializing bit indicates that the rail is ready to be enabled.

The Enabled bit represents the actual state of the rail and will be set whenever the rail is active and cleared whenever the rail is inactive.

The Fault bit will be set whenever the rail has experienced a fault condition. The type of fault is indicated by bit 18 (Overcurrent Fault) or bit 21 (Overtemperature Fault). Clear faults via the software API:

```
stem.rail[0].clearFaults()
```

The Hardware Configuration bits reflect the actual power path in use for `rail0` and are summarized in Table 14.

Config	Rail[0] Hardware Configruation Description
0	railOperationalStateLinear
2	railOperationalStateSwitcherLinear

Table 14: Rail[0] Hardware Configurations

RAIL0 and RAIL1 Current Limits

The current limit for each rail can be configured in software from 0A to 3A:

```
stem.rail[0].setCurrentLimit(microamps)
```

Note that the behavior following an overcurrent event differs between rails:

– `rail0` will simply reduce the output voltage to drive the specified current. No fault bits will be set in software.
– `rail1` will be turned off by the hardware if the output current goes above the set limit. The `rail1` Fault and Overcurrent Fault bits will be set and must be cleared before re-enabling the rail.

RAIL0 Overvoltage Protection

An overvoltage condition occurs when a voltage above the rail's setpoint is applied to the `rail0` output. When an overvoltage condition is detected, the hardware automatically disconnects the rail output. Once the overvoltage condition is removed, the rail output will resume desired operation without any software intervention. No fault bits will be set in software.

RAIL0 Temperature and Temperature Protection

The temperature of the adjustable rail (`rail0`) linear regulation stage is available via the API:

```
stem.rail[0].getTemperature(temperature)
```

Furthermore, an overtemperature condition will occur when the `rail0` temperature crosses the safe operating threshold, causing the rail to be automatically disabled. The `rail1` Fault and Overtemperature Fault bits will be set and must be cleared before re-enabling the rail.

RAIL0 Kelvin Sensing

Remote sensing can be applied to compensate for line loss in a system often found in high transient load applications. The MTM-PM-1 provides a "3-wire" interface to provide feedback to the MTM-PM-1 power supply to adjust appropriately and dynamically:

```
stem.rail[0].getKelvinSensingMode(bEnable)
stem.rail[0].setKelvinSensingMode(bEnable)
```

The bEnable parameter is an integer that correlates to the following:

- 0 (kelvinSensingOff)
- 1 (kelvinSensingOn)

Determine whether kelvin sensing is enabled or disabled. Kelvin sensing can be disabled if the power stage incurs a fault on the rail0 power stage.

```
stem.rail[0].getKelvinSensingState(state)
```

The state parameter is an integer that correlates to the following:

- 0 (kelvinSensingOff)
- 1 (kelvinSensingOn)

RAIL0 Power Enable Status

A pin which reports the state of the adjustable rail (`rail0`) is accessible from the MTM edge connector on pin 24, and switches when the rail is turned on or off. The status pin is in series with a diode, and as such is only capable of sourcing a V_{OH} logic level; it is not capable of sinking to ground. An external pull-down resistor may be required if reading the state of this pin.

The operational behavior of the pin is described as follows:

Rail[0] State	Power Enable Status
Off	Hi Impedance (Hi-Z)
On	Logic High (V_{OH})

Digital Entities

The MTM-PM-1 has two (2) digital input/outputs (DIO) controlled by the digital entity. Each DIO is controllable via software and is independently current limited for both source and sink currents.

All DIO are input and output capable:

```
stem.digital[0].setConfiguration(mode)
stem.digital[0].getConfiguration(mode)
```

The *mode* parameter is an integer that correlates to the following:

Value	Configuration
0	Input
1	Output
4	Hi Impedance (Hi-Z)
5	Input with Pull Down

Table 15: Digital IO Configuration Values

- Example: If a digital pin is configured as output mode, set the digital logic level:

```
stem.digital[0].setState(level)
```

Example: If a digital pin is configured as input mode, read the digital logic level:

```
stem.digital[0].getState(level)
```

If a digital pin is configured in Hi-Z mode its internal circuitry has been disconnected to create a high impedance. There are no functions that can act on this configuration.

Digital	Input	Output	HighZ	RCServo	Signal
DIO 0	Yes	Yes	Yes	None	None
DIO 1	Yes	Yes	Yes	None	None

Table 16: Digital IO Pin Configurations

I²C Entities

The MTM-PM-1 includes access to a single I²C bus operating at a set 1Mbit/s rate.

Note: The 1Mbit/s bus, while user-accessible, is also used for BrainStem network communication so there may be other, non-user-initiated traffic when other BrainStem modules are linked.

Example: Sending 2 bytes (0xABCD) through the I²C bus to a device with address 0x42:

```
stem.i2c.write(0x42, 2, 0xABCD)
```

Example: Reading 2 bytes of data from a device with address 0x42:

```
stem.i2c.read(0x42, 2, buffer)
```

Where *buffer* would be a char array in C++.

The maximum data size for individual read and write operations on an I²C bus through the BrainStem API is 20 bytes. Sending more than 20 bytes of information must be done as an iterated sequence.

Each I²C bus also includes 330Ω pull-up resistors on the SDA and SCL lines, disabled by default. When using the MTM-DAQ-2 in a linked system (communicating over the 1Mbit/s bus), only a single set of pull-ups along the bus should be enabled in order for the I²C bus to work properly (if more than one set is enabled, the lines cannot be pulled low for communication). Similarly, when using a single MTM device to communicate with an external device over the I²C bus, either the internal pull-ups can be enabled, or external hardware pull-ups added:

```
stem.i2c.setPullUp(bEnable)
```

The *bEnable* parameter is an integer that correlates to the following:

- 0 (I²C pull-ups off)
- 1 (i2cSetPullup)

Temperature Entities

An optional thermocouple interface can be ordered installed from the factory. Standard orders do not include this interface.

K-Type Thermocouple

If the MTM-PM-1 was purchased with the thermocouple interface installed, one can use a K-type thermocouple. The input from a connected and compatible K-type thermocouple can be read using the following function:

```
stem.temperature.getTemperature(temp)
```

Reflex RTOS

Reflex is Acroname's real-time operating system (RTOS) language which runs in parallel to the module's firmware. Reflex allows users to build custom functionality directly into the device. Reflex code can be created to run autonomously on the module or a host can interact with it through BrainStem's Timer, Pointer App and other entities.

Timer Entities

The Timer entity provides simple scheduling for events in the reflex system. The MTM-PM-1 includes 8 timers per reflex. Each timer represents a reflex definition to be executed upon expiration of a running timer. Timers can be controlled from a host, but the reflex code is executed on the device.

Example: Setting up and starting Timer 0 for single use:

```
stem.timer[0].setMode(timeModeSingle)
stem.timer[0].setExpiration(DELAY)
```

Reflex Definition: Timer 0 expiration callback:

```
reflex timer[0].expiration() { //Do Stuff }
```

Pointer Entities

Reflex and the Brainstem module share a piece of memory called the scratchpad which can be accessed via the Pointer Entity. The MTM-PM-1 has 4 pointers per reflex which allow access to the pad in a similar manner as a file pointer.

Example: Configure and access the scratchpad in static mode:

```
stem.pointer[0].setMode(pointerModeStatic)
stem.pointer[0].getByte(byte)
```

Reflex Pad: Single unsigned char definition:

```
pad[0:0] unsigned char byteValue
```

App Entities

Apps are reflex definitions that can be directly trigger by the host. These definitions are also capable of passing a parameter into or out of the app reflex definition. The MTM-PM-1 is equipped with 4 App Entities per reflex.

Example: Triggering App 0:

```
stem.app[0].execute(parameter)
```

Reflex Definition: App 0 callback:

```
reflex app[0](int appParam) { //Do Stuff }
```

MTM-PM-1 Supported Entity Methods Summary

Detailed entity class descriptions can be found in the BrainStem Reference[8]. A summary of MTM-PM-1 class options are shown below. Note that when using Entity classes with a single index (aka, 0), the index parameter can be dropped. For example:

```
stem.system[0].setLED(1) → stem.system.setLED(1)
```

Entity Class	Entity Option	Variable(s) Notes
digital[0-1]	setConfiguration	
	getConfiguration	
	setState	
	getState	
i2c[0]	write	
	read	
rail[0-1]	setEnable	
	setCurrentLimit	
	getCurrent	
	getCurrentLimit	
	getVoltage	
rail[0]	setVoltageSetpoint	
	getVoltageSetpoint	
	setOperationalMode	
	getOperationalMode	
	getOperationalState	
	getTemperature	
	getKelvinSensingEnable	
	setKelvinSensingEnable	
	getKelvinSensingState	
store[0-1]	getSlotState	
	loadSlot	
	unloadSlot	
	slotEnable	
	slotDisable	
	slotCapacity	
	slotSize	
system[0]	save	
	reset	
	setLED	
	getLED	
	setSleep	
	setBootSlot	
	getBootSlot	
	getInputVoltage	
	getVersion	
	getModuleBaseAddress	
	getModuleSoftwareOffset	

	setModuleSoftwareOffset	
	getModuleHardwareOffset	
	setHBInterval	
	getHBInterval	
	getRouterAddressSetting	
	getModule	
	getSerialNumber	
	setRouter	
	getRouter	
	getModel	
	routeToMe	
temperature[0]	getTemperature	
timer[0-7]	getExpiration	
	setExpiration	
	getMode	
	setMode	
Pointer[0-3]	getOffset	
	setOffset	
	getMode	
	setMode	
	getTransferStore	
	setTransferStore	
	initiateTransferToStore	
	initiateTransferFromStore	
	getChar	
	setChar	
	getShort	
	setShort	
	getInt	
	setInt	
App[0-3]	execute	

Table 17: Supported MTM-PM-1 BrainStem Entity API Methods[8]

[8]See BrainStem software API reference at https://acroname.com/reference/ for further details about all BrainStem API methods and information.

LED Indicators

The MTM-PM-1 board has a number of LED indicators to assist with MTM system development, debugging, and monitoring. These LEDs are shown in the diagrams below.

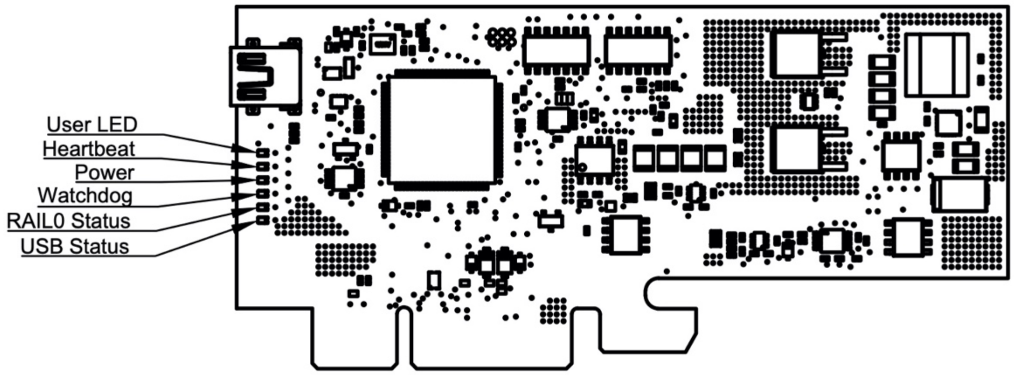


Figure 2: MTM-PM-1 LED Indicators

Edge Connector Interface

All MTM products are designed with an edge connector interface that requires a compatible board-edge connector on the carrier PCB. Acroname recommends the through-hole PCI-Express (PCIe) Vertical Connector. The connectors can be combined with an optional retention clip, as shown below. Representative part numbers are show in Table 18, and equivalent connectors are offered from a multitude of vendors.

Manufacturer	Manufacturer Part Number	Description
Amphenol FCI Samtec	10018784-10201TLF PCIE-064-02-F-D-TH	PCI-Express 64-position vertical connector
Amphenol FCI	10042618-003LF	PCI-Express Retention Clip (optional)

Table 18: PCI-Express Edge Connectors for MTM Products

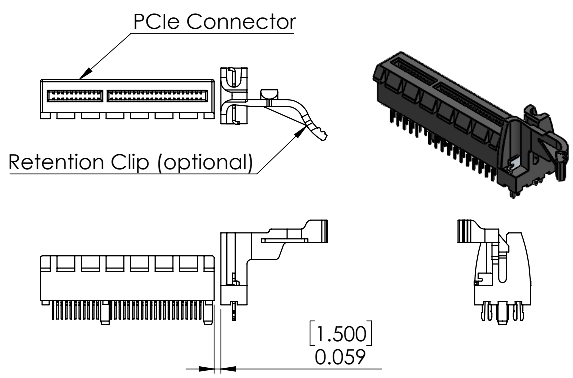


Figure 3: PCIe Vertical Connector with optional Retention Clip

MTM Edge Connector Specifications	Description
Contact Finish	Gold
Card Thickness	0.0625” [1.59mm]
Number of Rows	2
Number of Positions	Variable (see Table 18: PCI-Express Edge Connectors for MTM Products)
Pitch	0.039” (1.00mm)

Table 19: MTM Edge Connector Specifications

Mechanical

Dimensions are shown in mm. 3D CAD models are available through the MTM-PM-1 product page's Downloads section. A 3D CAD viewer with many different CAD formats available for download is available at https://a360.co/37Fhzhn.

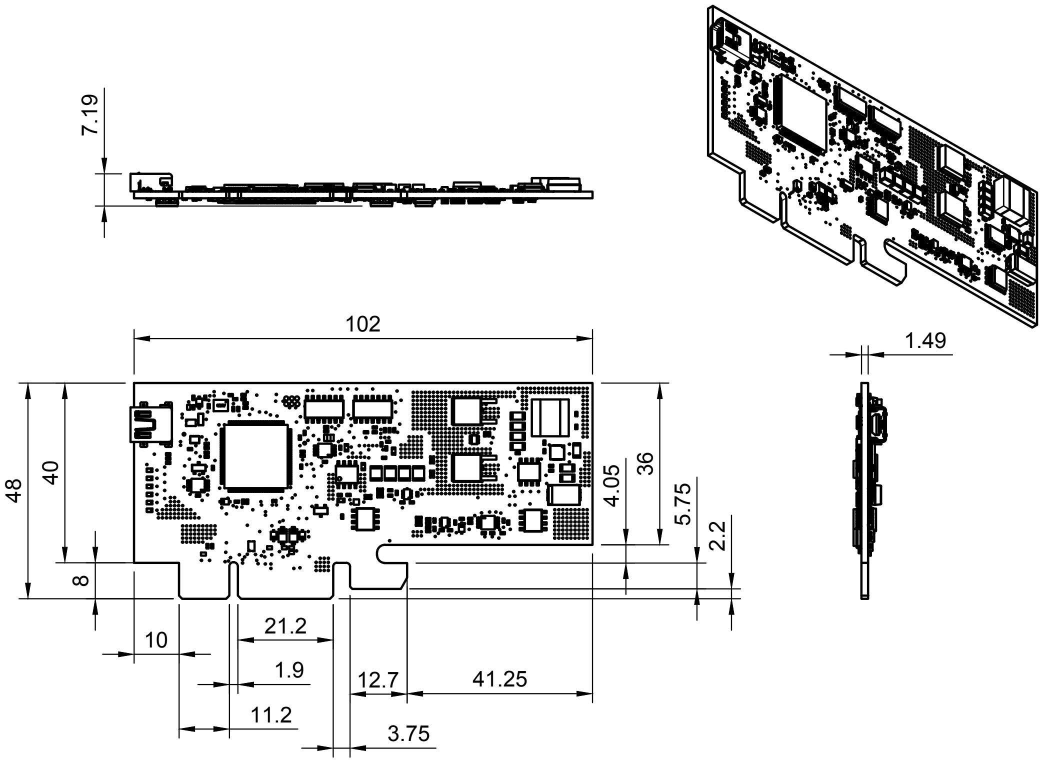


Figure 4: MTM-PM-1 Mechanical

Product Support

Questions about the product operation or specifications are welcome through Acroname's contact portals. Software downloads, reference API and application examples are available online at:

https://acroname.com/support

Direct communication and additional technical support are available at:

https://acroname.com/contact-us

2741 Mapleton Avenue

Boulder, CO, USA 80304-3837

720-564-0373 (phone)

Document Revision History

All major documentation changes will be marked with a dated revision code

Revision	Date	Engineer	Description
1.0	September 2015	JTD	Initial Revision
1.1	October 2015	JTD	Reformatted, added Entity Section Specifics
1.2	October 2015	MJK	Added pinout description
1.3	September 2016	JTD	Updated information
1.31	October 2016	LCD	Updated Overview, Features, Desription, added DO jitter
1.4	December 2016	JG	Clarified I2C pull-ups; update supported API calls
1.5	May 2017	RMN	Added missing Operational States
1.6	June 2020	ACRO	Included RAIL0 current range specification and Fusion 360 CAD link, formatting, entity updates.
1.7	July 2020	ACRO	Updated software control section, updated table numbers
1.8	September 2020	TDH	Rail API updates for Brainstem 2.8
1.9	February 2021	MJK	Contact information for technical support.
1.10	March 2022	CWG	Clarified module address table and Rail0 Power Enable Status behavior

Table 20: Document Revision History