Iono Pi Max v2 User Guide

October 2026

Revision 027

ICMX20XS *Iono Pi Max v2 Solo*
ICMX20XPL *Iono Pi Max v2 3+ Lite 1GB RAM*
ICMX20XP1 *Iono Pi Max v2 3+ 1GB RAM, 8GB eMMC*
ICMX20XP2 *Iono Pi Max v2 3+ 1GB RAM, 16GB eMMC*
ICMX20XP3 *Iono Pi Max v2 3+ 1GB RAM, 32GB eMMC*
ICMX20X4L *Iono Pi Max v2 4S Lite 1GB RAM*
ICMX20X41 *Iono Pi Max v2 4S 1GB RAM, 8GB eMMC*
ICMX20X42 *Iono Pi Max v2 4S 1GB RAM, 16GB eMMC*
ICMX20X43 *Iono Pi Max v2 4S 1GB RAM, 32GB eMMC*
ICMX20X4L4 *Iono Pi Max v2 4S Lite 4GB RAM*
ICMX20X434 *Iono Pi Max v2 4S 4GB RAM, 32GB eMMC*
ICMX20X438 *Iono Pi Max v2 4S 8GB RAM, 32GB eMMC*

SFERALABS

Safety information 6
Qualified personnel 6
Hazard levels 6
Safety instructions 7
General safety instructions 7
Battery 8
Introduction 9
Features 10
Device identification 12
Hardware setup 13
Connections 13
Opening the case 14
Closing the case 14
Raspberry Pi Compute Module installation 14
How to flash the eMMC 15
microSD installation 15
The external UPS battery 16
Replacing the RTC backup battery 16
Terminal blocks 18
Power supply 20
Battery charger 21
Auxiliary voltage outputs 21
PWR LED status 21
Software installation and configuration 22
Raspberry Pi OS Kernel module 22
Raspberry Pi OS Real Time Clock installation 22
Raspberry Pi OS SocketCAN driver 23
Raspberry Pi OS secondary SD interface 23
Raspberry Pi OS UART 24
Alternative set-up 24
Using Iono Pi Max 25
Real Time Clock 25
RS-232 serial port 25
RS-485 serial port 25
CAN FD port 27
EXP port 28

SD matrix 29
USB ports power management 31
ATECC608 secure element 32
EERAM 33
Push button 33
LEDs 33
Power supply 34
Shutdown enable mode 34
Shutdown wait time 35
Power-off time 35
Power-up delay 35
Power-up mode 35
UPS and battery charger 35
Auxiliary voltage outputs 36
Hardware watchdog 36
Internal fan and temperature sensors 37
Analog inputs 37
AVx: Analog voltage inputs 38
AIx: Analog current inputs 38
ATx: Temperature sensors inputs 39
AOx: Analog voltage and current outputs 39
DIx: Digital inputs 39
OCx: Open collector outputs 39
DTx: Digital input/output (Wiegand, 1-Wire) 40
Ox: Relays 41
Earthquake sensor module 41
Advanced usage 42
Dedicated GPIO pins 42
Raspberry Pi I2C bus addresses 43
I2C Configuration and control registers map 44
Firmware upload and MCU programming 59
Thermal considerations for Iono Pi Max 62
Block diagram 64
Technical specifications 68
Dimensions 73
Disposal 74

Installation and use restrictions 74
Standards and regulations 74
Safety instructions 74
Set-up 74
Conformity Information 75
EU 75
USA 75
CANADA 76
RCM AUSTRALIA / NEW ZEALAND 76

Be sure to always remove the power supply before inserting or removing the Raspberry Pi Compute Module from the Iono Pi Max.

Follow all applicable electrical safety standards, guidelines, specifications and regulations for installation, wiring and operations of Iono Pi Max.

Carefully and fully read this Iono Pi Max user guide before installation.

Iono Pi Max is not authorised for use in safety-critical applications where a failure of the product would reasonably be expected to cause personal injury or death. Safety-critical applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Iono Pi Max is neither designed nor intended for use in critical military or aerospace applications or environments and for automotive applications or environment. Customer acknowledges and agrees that any such use of Iono Pi Max is solely at Customer's risk, and that Customer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

Sfera Labs S.r.l. may make changes to specifications and product descriptions at any time, without notice. The product information on the web site or materials is subject to change without notice.

Please download and read the Sfera Labs Terms and Conditions document available at:

http://www.sferalabs.cc

Iono and Sfera Labs are trademarks of Sfera Labs S.r.l. Other brands and names may be claimed as the property of others.

Copyright © 2023-2025 Sfera Labs S.r.l. All rights reserved.

Safety information

Carefully and fully read this user guide before installation and retain it for future reference.

Qualified personnel

The product described in this manual must be operated only by personnel qualified for the specific task and installation environment, in accordance with all relevant documentation and safety instructions. A qualified person should be capable of fully identifying all installation and operation risks and avoid potential hazards when working with this product.

Hazard levels

This manual contains information you must observe to ensure your personal safety and prevent damage to property. Safety information in this manual are highlighted by the safety symbols below, graded according to the degree of danger.

DANGER

Indicates a hazardous situation which, if not avoided, **will** result in death or serious personal injury.

WARNING

Indicates a hazardous situation which, if not avoided, **may** result in death or serious personal injury.

CAUTION

Indicates a hazardous situation which, if not avoided, can result in minor or moderate personal injury.

NOTICE

Indicates a situation which, if not avoided, can result in damage of property.

Safety instructions

General safety instructions

Protect the unit against moisture, dirt and any kind of damage during transport, storage and operation. Do not operate the unit outside the specified technical data.

Never open the housing. If not otherwise specified, install in closed housing (e.g. distribution cabinet). Earth the unit at the terminals provided, if existing, for this purpose. Do not obstruct cooling of the unit. Keep out of the reach of children.

WARNING

Life threatening voltages are present within and around an open control cabinet.

When installing this product in a control cabinet or any other areas where dangerous voltages are present, always switch off the power supply to the cabinet or equipment.

WARNING

Risk of fire if not installed and operated properly.

Follow all applicable electrical safety standards, guidelines, specifications and regulations for installation, wiring and operations of this product.

The Raspberry Pi Compute Module board could generate a substantial amount of heat when the software forces the CPU and/or GPU to operate at high load levels. Ensure that the product is properly installed and ventilated to prevent overheat.

The Iono Pi Max internal power supply could generate a substantial amount oh heat, particularly when subject to a significant amount of electrical load.

An internal fan significantly improves the airflow and heat dissipation. Depending on external environment conditions, the fan could collect a significant amount of dust or other impurities, that could prevent it from spinning or could reduce its effectiveness. Periodically check that the fan is not blocked or partly obstructed.

NOTICE

The connection of expansion devices to this product may damage the product and other connected systems, and may violate safety rules and regulations regarding radio interference and electromagnetic compatibility.

Use only appropriate tools when installing this product. Using excessive force with tools may damage the product, alter its characteristics or degrade its safety.

Battery

Iono Pi Max uses a small lithium non-rechargeable battery to power its internal real time clock (RTC).

WARNING

Improper handling of lithium batteries can result in an explosion of the batteries and/or release of harmful substances.

Worn-out or defective batteries can compromise the function of this product.

Replace the RTC lithium battery before it is completely discharged. The lithium battery must be replaced only with an identical battery. See the "Replacing the RTC backup battery" section for instructions.

Do not throw lithium batteries into fire, do not solder on the cell body, do not recharge, do not open, do not short-circuit, do not reverse polarity, do not heat above 85°C and protect from direct sunlight, moisture and condensation.

Dispose of used batteries according to local regulations and the battery manufacturer's instructions.

Introduction

Iono Pi Max is an extremely versatile industrial server with a huge variety of Input/Output lines and standard communication interfaces, based on the Raspberry Pi Compute Module, suitable for use in professional and industrial applications where reliability and service continuity are key requirements, and housed in a compact 6 modules DIN rail case.

Iono Pi Max can be used for an incredible range of applications, from data acquisition and control, to home and building automation, as well as access control, hotel room control solutions, environmental monitoring and many others.

It is available in several versions. Iono Pi Max 3+ ships with a Raspberry Pi Compute Module 3+ pre-installed. Iono Pi Max 4S ships with a Raspberry Pi Compute Module 4S pre-installed. Iono Pi Max Solo doesn't include the Compute Module and is compatible with all current versions of the Compute Module: 3, 3 Lite, 3+, 3+ Lite, 4S and 4S Lite.

The information contained in this manual covers both Iono Pi Max 3+/4S and Iono Pi Max Solo. All Iono Pi Max versions are identical, except for the Compute Module model installed.

This manual generally refers to all products as Iono Pi Max.

IONO PI MAX

Features

Iono Pi Max, all versions:

- ✓ 10-50Vdc power supply, with surge and reverse polarity protection, and 3.3A resettable fuse
- ✓ integrated uninterruptible power supply (UPS), with external lead-acid 12V or 24V battery (with reverse polarity, surge and over-current protection)
- ✓ software-controlled auxiliary 12-24Vdc stabilised supply output, to power external devices through Iono Pi Max UPS (with reverse current and surge protection)
- ✓ software-controlled auxiliary 5Vdc stabilised supply output
- ✓ voltage and current monitoring of power supply input, battery and auxiliary supply output
- ✓ highly accurate real time clock with temperature compensated crystal oscillator (TCXO) and replaceable CR1220 Lithium / Manganese Dioxide back-up battery
- ✓ standard RS-232 interface to the Raspberry Pi serial line, up to 115.200 bps
- ✓ standard RS-485 interface, routed to the Raspberry Pi serial line or USB, with opto-isolator and electrostatic discharge protection, up to 115.200 bps
- ✓ Controller Area Network (CAN) V2.0B and CAN-FD interface, support speeds up to 8 Mb/s, with opto-isolator and electrostatic discharge protection
- ✓ 5V-level, protected, I2C expansion bus
- ✓ 4 power relay outputs rated for 6A at 250V, with NO and NC contacts
- ✓ 4 open-collector outputs
- ✓ 2 0-10V or 0-20mA analog outputs
- ✓ 4 analog voltage inputs 0-20V
- ✓ 4 analog current inputs 0-20mA
- ✓ 2 Pt100 or Pt1000 temperature sensor inputs
- ✓ 4 digital input lines
- ✓ 4 digital 5V-level I/O lines, with 1-Wire and Wiegand support
- ✓ 10/100 Ethernet port
- ✓ two USB 2.0 type-A ports, with individual power control and fault detection
- ✓ one internal USB 2.0 type-A port, to embed USB dongles inside the enclosure
- ✓ dual microSD card architecture. Iono Pi Max routes its two microSD card holders to the Compute Module through a high-speed switching matrix, controlled by the on-board microcontroller. Each SD card can be set as the boot card, with the other available as additional storage. This enables for separate OS/data storage, SD redundancy and in-field full-system upgrades. The microSD slots are hidden, and can be accessed opening the case
- ✓ hardware watchdog for independent monitoring of the Raspberry Pi Compute Module. Controlled via GPIO pins and capable of performing a full power-cycle of the Compute Module. The hardware watchdog can also control the microSD switch matrix, swapping the boot card in case the watchdog is triggered

- ✓ powerful and user-programmable 32 bit microcontroller (ATSAME54), with 1 MB program memory and 256 KB RAM. The microcontroller is interconnected to the Raspberry Pi via I2C, USB and CAN. It can also take direct low-level control of all serial ports and digital as well as analog I/O
- ✓ integrated Serial Wire Debug (SWD) interface to the Iono Pi Max microcontroller, allowing in-field firmware upgrades, but also development and debugging, directly from the Raspberry Pi
- ✓ embedded Microchip ATECC608 secure element chip
- ✓ front-panel RGB LEDs for power supply and serial line activity, can also be directly controlled by the Raspberry Pi
- ✓ on-board piezoelectric buzzer, for acoustic feedback
- ✓ front-panel push-button, connected to a GPIO pin
- ✓ power supply, serial and I/O connections on pluggable terminal blocks
- ✓ internal temperature sensors for system temperature monitoring
- ✓ internal fan, controlled by the microcontroller based on current load and internal temperature
- ✓ packaged in a compact 6 modules DIN rail case
- ✓ optional earthquake sensor module

Iono Pi Max v2 Solo:

- ✓ compatible with Raspberry Pi Compute Module 3, 3+, and 4S

Iono Pi Max v2 3+:

- ✓ pre-installed and tested Raspberry Pi Compute Module 3+

Iono Pi Max v2 4S:

- ✓ pre-installed and tested Raspberry Pi Compute Module 4S

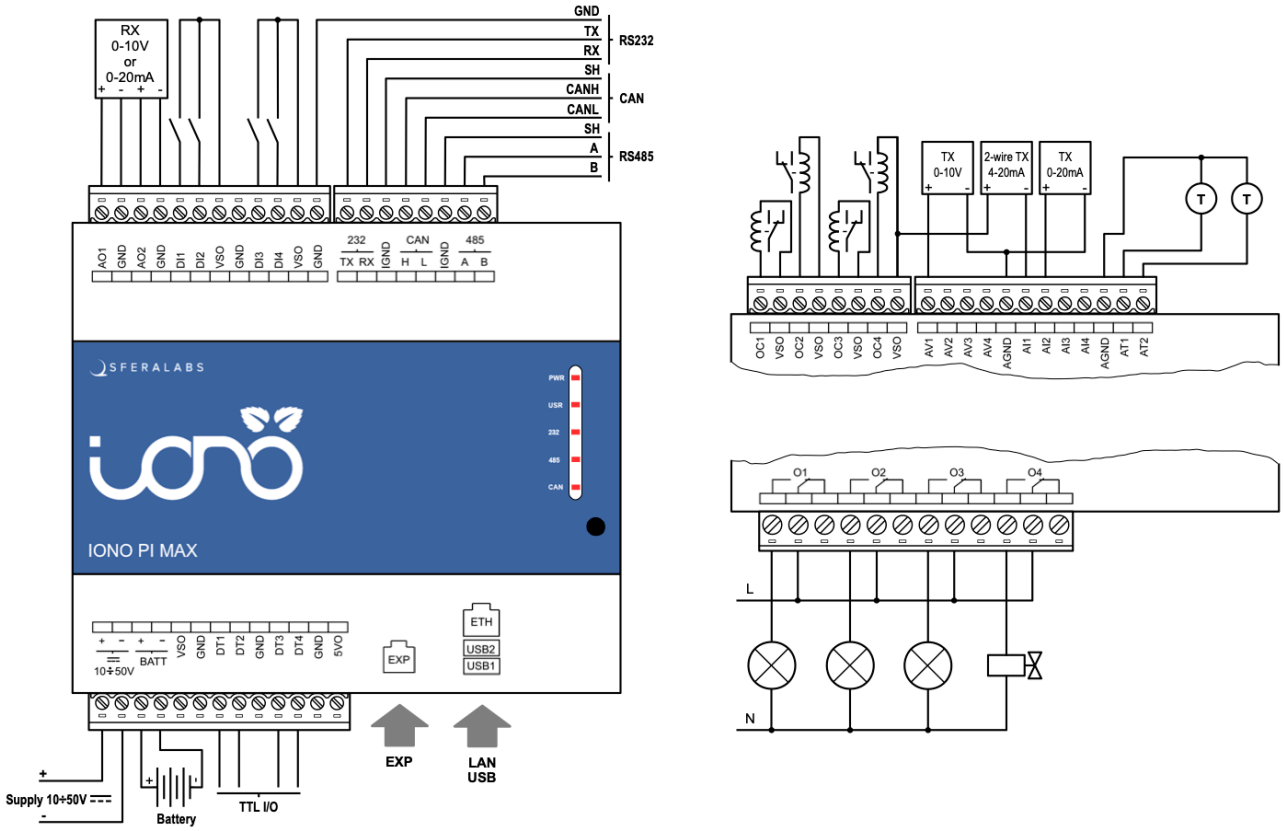
Device identification

The device can be identified with the information provided in the rating and identification plate, permanently attached to the side of the case.

EXAMPLE RATING AND IDENTIFICATION PLATE

Hardware setup

Connections



IONO PI MAX CONNECTION EXAMPLE

For the initial set-up, the plastic DIN rail enclosure must be removed to access the circuit boards and internal connectors, to install or remove the Raspberry Pi Compute Module and the μSD cards.

The case also needs to be opened if the factory configuration of the RS-485 and CAN termination resistors must be changed (see below), or to connect to the microcontroller programming and debug header.

NOTICE

Before opening the Iono Pi Max enclosure, disconnect all power sources and any connection to external devices, including USB and Ethernet cables.

Opening the case

Follow these steps, in the exact order, to open the case:

1. Remove the green pluggable terminal blocks
2. Remove the black plastic DIN rail hook
3. With a small flat screwdriver gently separate the case bottom and boards assembly from the top shell; lift both sides gradually; pay attention not to touch the circuit boards with the screwdriver's tip
4. The circuit boards assembly will simply slide out of the case, as it is mechanically held in place by the case bottom only.

Closing the case

5. Remove the front panel. This will help aligning the circuit boards assembly with the case walls
6. Replace the circuit boards assembly inside the case; ensure that the USB/Ethernet connectors and terminal blocks are aligned with the openings in the case
7. Replace the case bottom, partially inserting both sides, then gently push on both sides. The case bottom must perfectly fit and firmly hook to the case cover. If it doesn't, the circuit boards assembly is not perfectly fit into the case top shell
8. Replace the front panel, ensuring that the frontal push-button is aligned to the panel's hole
9. Replace the black plastic DIN rail hook.

OPENING AND CLOSING THE CASE

Raspberry Pi Compute Module installation

To install your own Compute Module in Iono Pi Max Solo, or replace the pre-installed module, no tools are required once the plastic enclosure has been removed and the circuit boards assembly is exposed. The Compute Module fits in a standard 200 pin DDR2 SODIMM (JEDEC MO-224 mechanical specification) slot.

NOTICE

Static electricity can damage the components in your system. To protect your system's components from static damage during the installation process, touch any of the unpainted metal surfaces on your computer's frame or wear an ESD wrist strap before handling internal components. Either method will safely discharge static electricity that's naturally present in your body.

When handling the Raspberry Pi Compute Module, be sure to hold it along the side edges using your thumb and index finger. Avoid touching the components and pin connectors as damage can occur.

Align the notches on the Compute Module with the ridges in the DDR2 slot. Insert the module using firm and even pressure, and push down on the module until the clips on the side of the slot snap into place.

How to flash the eMMC

In order to flash the Compute Module's eMMC memory, an additional board, like the *Compute Module IO Board V3*, shall be used. Use the latest RPIBOOT utility from Raspberry Pi and refer to its documentation[1] for installation and usage details.

Alternatively, if using the Compute Module 4S (or a Compute Module 3/3+ with *USB host boot mode* previously enabled[2]) you can boot the unit from USB[3], access via SSH, and flash the eMMC which will be available as a local mass storage device.

microSD installation

Iono Pi Max has two microSD slots to install flash memory cards.

When the Compute Module Lite is installed, as it lacks an internal eMMC chip, one of the two microSD slots will be used as the boot SD card, just like with a Raspberry Model B board. The other microSD can be used as additional storage. Dedicated commands to the microcontroller allow to select which slot is used for boot, and if the other is enabled or not.

When a Compute Module with embedded eMMC is installed, both microSD slots can be used as additional storage. You can only access one SD card at a time, but the active card can be selected sending the appropriate commands to the microcontroller.

The microSD slots are not directly accessible with the case closed. The slots are located on the vertical board, next to the USB/Ethernet modular connector.

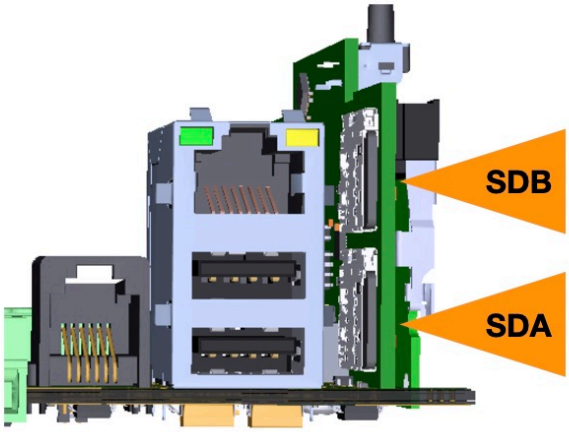
The bottom slot is SDA, and the top is SDB.

---

[1] https://www.raspberrypi.com/documentation/computers/compute-module.html#flashing-the-compute-module-emmc

[2] https://www.raspberrypi.com/documentation/computers/raspberry-pi.html#usb-host-boot-mode

[3] https://www.raspberrypi.com/documentation/computers/raspberry-pi.html#booting-from-usb-mass-storage



MICRO SD SLOTS

The external UPS battery

To take advantage of the UPS features of Iono Pi Max, you can connect an external lead acid rechargeable battery to the BATT terminals. See the connection diagram above.

WARNING

Improper handling of lead acid batteries can result in an explosion of the batteries and/or release of harmful substances.

Worn-out or defective batteries can compromise the function of this product.

Replace the battery every 5 years even if the battery is still working properly.

Only use a lead acid battery with electrical ratings recommended in the technical specifications for this product (12V or 24V, 7Ah max).

Dispose of used batteries according to local regulations and the battery manufacturer's instructions.

Follow the battery manufacturer's instructions when installing the external UPS battery (not provided).

Replacing the RTC backup battery

Iono Pi Max has a hardware real time clock with a dedicated long-life non-rechargeable back-up battery.

Iono Pi Max is shipped with a **CR1220**[4] Lithium / Manganese Dioxide (Li/MnO2) battery installed.

[4] BR1220 batteries (Lithium Poly-carbonmonofluoride) can also be used, for improved shelf and service life.

The battery is only used to power the RTC chip when the main power is not available. Depending on operating conditions it should last three or more years[5] if the Iono Pi Max is not powered, or up to the battery's lifetime if it regularly receives external power.

The RTC backup battery holder is on the vertical circuit board, opposite the Compute Module slot. In the Iono Pi Max, the battery is not accessible from the outside. You should first remove the case body to gain access to the Iono Pi Max circuit boards assembly. Follow the procedure described at the beginning of this chapter to open and replace the case.

WARNING

Improper handling of lithium batteries can result in an explosion of the batteries and/or release of harmful substances.

Worn-out or defective batteries can compromise the function of this product.

KEEP OUT OF REACH OF CHILDREN. Swallowing may lead to serious injury or death in as little as 2 hours due to chemical burns and potential perforation of the esophagus. Immediately see doctor.

Replace the RTC lithium battery before it is completely discharged. Replace the battery every 5 years even if the battery is still working properly. The lithium battery must be replaced only with an identical **CR1220** or **BR1220** battery.

Do not throw lithium batteries into fire, do not solder on the cell body, do not recharge, do not open, do not short-circuit, do not reverse polarity, do not heat above 85°C and protect from direct sunlight, moisture and condensation.

Dispose of used batteries according to local regulations and the battery manufacturer's instructions.

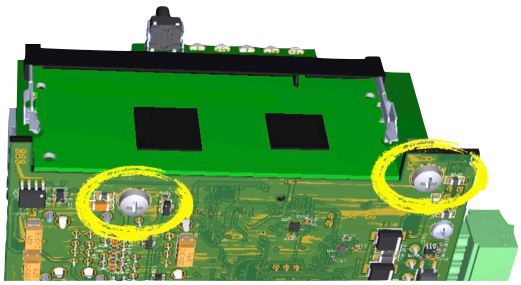
NOTICE

Before opening the Iono Pi Max enclosure, disconnect all power sources and any connection to external devices, including USB cables.

Once the circuit boards assembly is extracted from the enclosure, the most convenient way to access the battery holder is to first separate the vertical board from the base board.

This is easily accomplished by unscrewing the two screws on the bottom side of the base board, that secure the vertical board to it. Pay attention not to touch any of the electronic components with the screwdriver head. Once the screws are removed, the vertical board easily separates from the base. Be careful not to stress the board-to-board connector.

[5] Depending on environmental conditions.

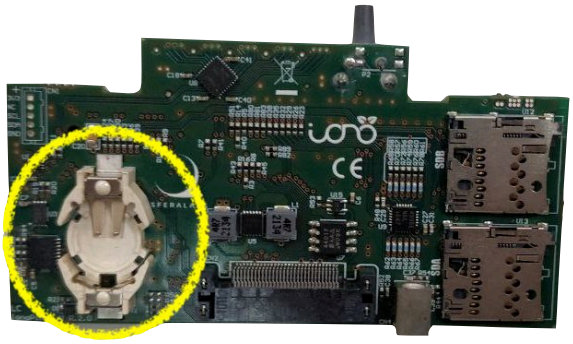


VERTICAL BOARD SCREWS

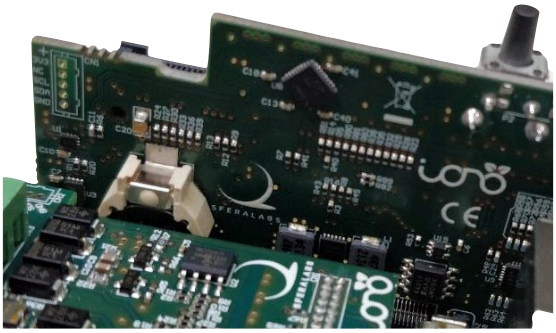
Use a non-conductive pin or small tool to help extract the battery from its holder. Insert the new battery with a gentle push. You don't need tools to insert the battery. The battery is held into place by a spring contact.

Be sure to insert the battery so that the negative (-) terminal is facing the vertical circuit board, and touches the contact pad at the center of the battery holder, as shown in the photo below. Reattach the vertical board to the base; use light torque to tighten the screws.

The real time clock will reset its time immediately when the RTC backup battery is disconnected.



RTC BACKUP BATTERY HOLDER



RTC BACKUP BATTERY INSTALLED

Terminal blocks

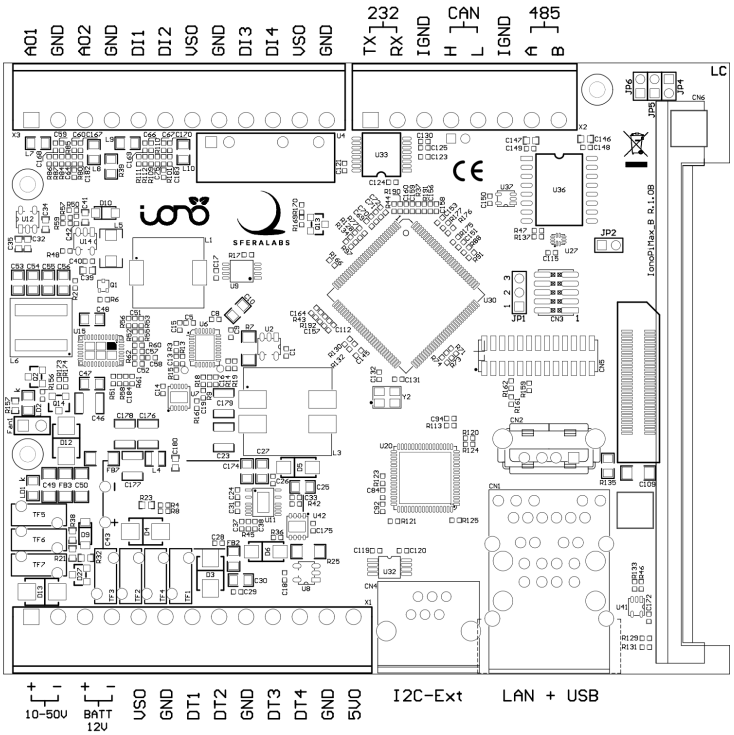
Iono Pi Max has several pluggable terminal blocks.

All power supply, input/output and serial interfaces are on 3.81mm pitch terminal blocks.

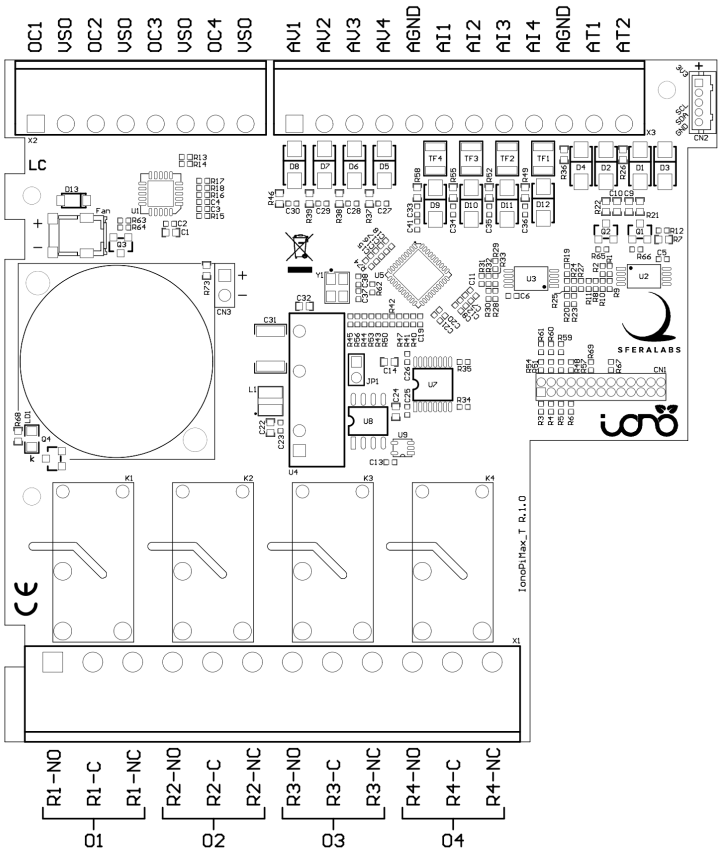
The maximum conductor cross section is 1.31 mm² (16 AWG), or 0.5 mm² when using ferrules (highly recommended). Recommended stripping length is 6 mm. Screw thread is M2. Never exceed 0.3 Nm torque when tightening the screws.

Two 6 positions terminal blocks are dedicated to the power relays outputs.

The maximum conductor cross section is 3.31 mm² (12 AWG). Recommended stripping length is 7 mm. Screw thread is M3. Never exceed 0.5 Nm torque when tightening the screws.



THE IONO PI BOTTOM CIRCUIT BOARD WITH TERMINAL BLOCKS



THE IONO PI TOP CIRCUIT BOARD WITH TERMINAL BLOCKS

Power supply

Iono Pi Max can be powered with DC voltage only:

✓ DC: nominal voltage range 10.0 V to 50.0 V, 3 A max input current.

Respect the correct polarity shown in the schematic diagram (+ -). The power supply circuit implements reverse polarity protection using auto resetting fuses and surge protection. The resetting fuse threshold is 3.3 A at 20 C ambient temperature.

The PWR color LED, visible through the front panel of the DIN-rail case, is lit when power supply is available.

The Iono Pi Max power supply stage is optimised to deliver up to 30W total power through its 5 V main regulator. This regulator supplies all internal voltages, including the battery charger voltage, but not the VSO (auxiliary regulated 12-24 Vdc output) regulator voltage, that draws directly from VIN or battery and can deliver up to 8 W.

When the input voltage is below 24 V, if the total power demand is very high from both the main regulator and the VSO output, the input current threshold limit could be exceeded, causing the resettable fuse to trip.

Battery charger

The Iono Pi Max UPS (uninterruptible power supply) implements a programmable battery charger, that can charge an external lead-acid battery (other chemistries could be supported with future firmware releases).

The battery line is protected by an internal 3.3 A resettable fuse.

You can use either 12 V or 24 V batteries. The supported maximum battery capacity is 60 Ah, but the battery charger can deliver no more than approximately 600 mA charging current, so charging time for large batteries could be very long.

For low load applications, a 12 V 1.2 Ah battery is sufficient to keep Iono Pi Max running for several minutes and perform an orderly shutdown. A 12 V or 24 V 7 Ah battery is recommended for high load applications.

The factory default configuration for the battery charger has the charger enabled and configured for a 12 V lead-acid battery with 800mAh capacity.

WARNING

Ensure that the charger parameters are correctly configured before connecting the battery to the battery terminals.

Auxiliary voltage outputs

Iono Pi Max has an auxiliary regulated voltage output, labeled VSO on the terminal block, that can be set in a 12 Vdc to 24 Vdc voltage range, and is powered by the UPS. It can deliver up to 8 W, and its output is protected by a 1.1 A resettable fuse.

Iono Pi Max also has an auxiliary 5 Vdc voltage output pin on the terminal block, labeled 5VO, to supply power to external 5 V loads, like 1-Wire devices. The current output limit is 500 mA. An internal protection circuit shuts down the 5VO line if the current limit is exceeded.

In the factory default configuration, both the 5VO and VSO outputs are enabled, with VSO set for 12 V output voltage.

PWR LED status

During normal operation, the PWR LED is stable green or stable blue (base color) to indicate that the Raspberry Pi is using respectively SDA or SDB as boot SD card.

During power off time in a power cycle, the LED is stable red.

When the shutdown delay is running, a red blink is alternated with the base color.

When the watchdog timeout expires, a yellow blink is alternated with the base color.

When the module is running on battery power, the base color blinks on/off.

When system status warnings are active, a white blink is alternated with the base color.

In concurrent situations, priorities are given in the above order.

Software installation and configuration

All features of Iono Pi Max can be used with any operating system, programming language or framework compatible with Raspberry Pi.

The following sections describe how to install utility drivers for the Raspberry Pi OS (formerly Raspbian) or to access its features from alternative set-ups.

Raspberry Pi OS Kernel module

The Iono Pi Max Kernel module can be used to easily access all of Iono Pi Max features via a sysfs file system.

Refer to the following instructions to download and install the module or go to:

https://github.com/sfera-labs/iono-pi-max-kernel-module

for updated instructions and further details.

Download and install:

```
$ sudo apt install git raspberrypi-kernel-headers
$ git clone --depth 1 https://github.com/sfera-labs/iono-pi-max-kernel-module
$ cd iono-pi-max-kernel-module
$ make
$ sudo make install
$ dtc -@ -Hepapr -I dts -O dtb -o ionopimax.dtbo ionopimax.dts
$ sudo cp ionopimax.dtbo /boot/overlays/
```

Add the following line to /boot/config.txt:

```
dtoverlay=ionopimax
```

Optionally, to be able to use the /sys/ files not as super user, create a new group "ionopi" and set it as the module owner group by adding an udev rule:

```
$ sudo groupadd ionopimax
$ sudo cp 99-ionopimax.rules /etc/udev/rules.d/
```

and add your user to the group, e.g., for user "pi":

```
$ sudo usermod -a -G ionopimax pi
```

Reboot:

```
$ sudo reboot
```

You will find all the available devices under the directory /sys/class/ionopimax/ and you will be able to control all the exposed functionalities by simply reading/writing the corresponding files from your application or any tool or framework that supports file access.

Refer to the next chapter for usage details about each functionality.

Raspberry Pi OS Real Time Clock installation

The RTC is based on the NXP PCF2131 RTC module and is connected to the Raspberry Pi via its I²C serial bus.

Refer to the following instructions to download and install the module or go to:

https://github.com/sfera-labs/rtc-pcf2131

for updated instructions.

Download and install:

```
$ sudo apt install git raspberrypi-kernel-headers
$ git clone --depth 1 https://github.com/sfera-labs/rtc-pcf2131
$ cd rtc-pcf2131
$ make
$ sudo make install
$ dtc -@ -Hepapr -I dts -O dtb -o rtc-pcf2131.dtbo rtc-pcf2131.dts
$ sudo cp rtc-pcf2131.dtbo /boot/overlays/
```

Add the following line to /boot/config.txt:

```
dtoverlay=rtc-pcf2131
```

Reboot:

```
$ sudo reboot
```

Raspberry Pi OS SocketCAN driver

To enable SocketCAN support for the CAN FD/CAN 2.0 controller (MCP2518FD) of Iono Pi Max, add to /boot/config.txt the following line:

```
dtoverlay=mcp251xfd,spi0-0,interrupt=28
```

Reboot:

```
$ sudo reboot
```

After reboot you should see a new "can0" interface available:

```
$ ifconfig -a
...
can0: flags=193<NOARP>  mtu 16
...
```

You can now use the SocketCAN framework to control the CAN interface.

Following some examples to enable and test CAN FD communication.

Initialize "can0" specifying communication parameters and enabling the CAN FD protocol:

```
$ sudo ip link set can0 up type can bitrate 1000000 dbitrate 8000000 restart-ms
1000 berr-reporting on fd on
$ sudo ifconfig can0 txqueuelen 65536
```

Dump data from the bus:

```
$ candump can0
```

Generate random traffic:

```
$ cangen can0 -mv
```

Raspberry Pi OS secondary SD interface

The secondary SD interface, SD1, must be enabled adding the following line in the /boot/config.txt file:

```
dtoverlay=sdio,bus_width=4,poll_once=off
```

Raspberry Pi OS UART

By default, the Linux serial console is active on the main UART (/dev/ttyAMA0). If you are using the RS-232 or RS-485 interfaces connected to the main UART (see next chapter), you should disable it to avoid console data being sent to the serial port.

To disable the ttyAMA0 console service, run:

```
$ sudo systemctl disable serial-getty@ttyAMA0.service
```

and edit the /boot/cmdline.txt file to delete the serial console configuration:

```
console=serial0,115200
```

Alternative set-up

If you are using other Operating Systems or need not to use the provided drivers, you can access all the functionalities of Iono Pi Max via dedicated GPIO pins, the I²C bus and the SPI interface.

All the features directly controlled via GPIO, such as digital inputs and watchdog control, can be accessed using any of the standard methods provided by the operating system or any available library for your programming language or framework of choice.

Make sure the GPIO pins are correctly configured as inputs or outputs, depending on the function and no pull-up/pull-down is enabled.

All features of Iono Pi Max controlled via I²C are accessible at the address 0x35 on bus 1 ("i2c-1" - GPIO2/SDA1 and GPIO3/SCL1).

The RTC (PCF2131) is directly accessible at address 0x53 on bus 1.

The CAN controller (MCP2518FD) is connected to the SPI interface chip select 0 ("spi0.0" - GPIO8-11) and the interrupt signal (INT) to GPIO28.

Refer to the next chapter for details about each feature.

Using Iono Pi Max

Most of the features discussed in this chapter are based on features implemented on our firmware code running on the SAME54 microcontroller.

The microcontroller can be programmed with user developed custom firmware, that could replace the factory installed version. The "Firmware upload and MCU programming" chapter below shows how to install and configure the programming tools on Raspberry Pi OS or other similar Linux distributions.

WARNING

The microcontroller firmware implements critical logic and checks to ensure safe and reliable operations of Iono Pi Max including, but not limited to, fan control for heat dissipation, the battery charger logic, the VSO regulator control, watchdog, μSD switch matrix control, power distribution voltages, currents and internal temperature monitoring.

In order to develop a safe and reliable custom firmware, you should be fully aware of the internal hardware architecture of Iono Pi Max, and of all safety and control requirements needed for your application and installation constraints.

Real Time Clock

Iono Pi Max has a hardware real time clock with a dedicated long-life non-rechargeable back-up battery.

Once the RTC driver is properly installed, you will simply use the date and time commands to control the hardware clock.

Note that, while by default the RTC chip is connected to the Pi's I2C bus, it could be re-routed, using custom firmware, to the SAME54 I2C bus (DMZ_I2C).

RS-232 serial port

Iono Pi Max uses the Raspberry Pi standard UART TX/RX pins on the GPIO connector to implement a standard RS-232 port, without hardware flow control. This port is not galvanically isolated. The Pi main UART is accessed as /dev/ttyAMA0 in Raspberry Pi OS.

The RS-232 port can be alternatively routed to the microcontroller and used locally by custom firmware, or made available to the Raspberry Pi as a USB serial device (/dev/ttyACM0), instead of the RS-485, using the serial/rs232_rs485_inv sysfs kernel module file (I2C control register 26).

RS-485 serial port

Iono Pi Max implements a standard RS-485 port, routed to the microcontroller and made available to the Raspberry Pi as a USB serial device (/dev/ttyACM0).

The RS-485 port is galvanically isolated using a dedicated isolated differential line transceiver and a high-efficiency DC-DC converter. This configuration should prevent ground loops between devices connected through this port.

The IGND terminal is the isolated ground reference for the RS-485 serial lines. Always use this line, not GND, for the RS-485 connection.

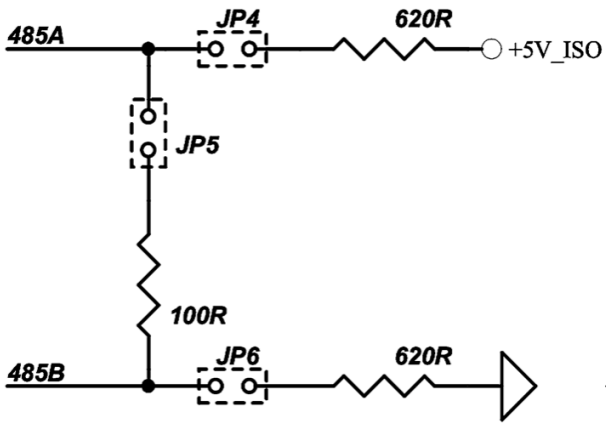
The RS-485 TX/RX switching is implemented automatically in the Iono Pi Max microcontroller, based on the USB serial device speed and number of bits settings, and is completely transparent to the software on Raspberry Pi.

The RS-485 port can be alternatively used locally by custom firmware running on the microcontroller, or routed to the Raspberry Pi standard UART TX/RX pins on the GPIO connector, instead of the RS-232, using the serial/rs232_rs485_inv sysfs kernel module file (I2C control register 26). When the RS-485 is routed to the Pi's UART, the RS-485 TX/RX switching requires explicitly setting the serial configuration parameters or manually controlling the TX-enable line via serial/rs485_txe sysfs file (I2C control register 27). If not configured nor manually controlled it will work in passive mode.

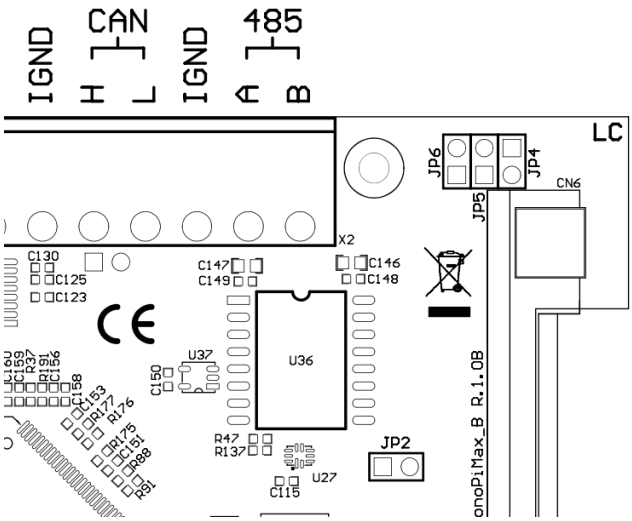
The RS-485 line has 620 Ohm pull-up and pull-down resistors on lines A and B. The biasing resistors ensure that, during idle periods the data lines are kept at a stable voltage level and prevent false triggering of receiver input. These resistors can be disabled by removing jumpers JP4 and JP6 respectively.

A 100 Ohm termination resistor between A and B is connected when the JP5 jumper is inserted.

Jumpers JP4, JP5 and JP6 are located on the base circuit board, to the left of the RS-485 terminal block.



RS-485 JUMPERS SCHEMATIC



RS-485 JUMPERS AND SWITCH POSITION

CAN FD port

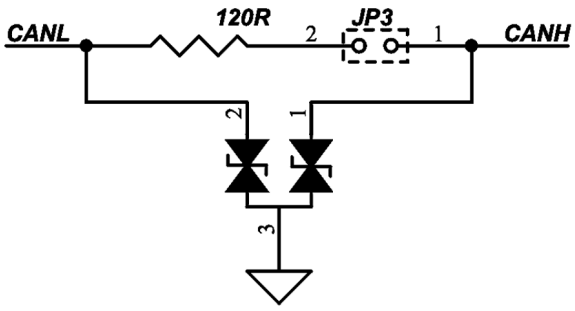
The CAN interface is based on the Microchip MCP2518FD CAN FD Controller and the Microchip MCP2558FD high-speed CAN FD transceiver. They implement both CAN 2.0B and CAN FD modes, with support for up to 8 Mbps data rate and 1 Mbps arbitration rate.

The CANH and CANL lines are available on the terminal block.

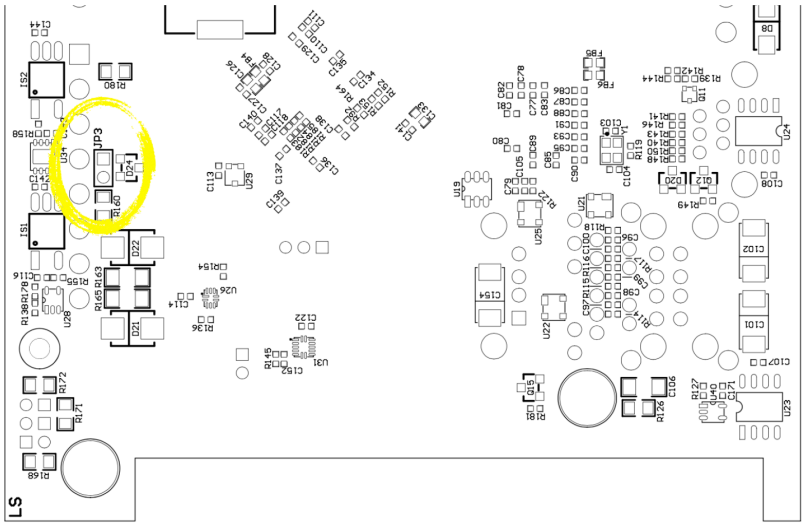
The CAN transceiver's TX and RX lines are isolated from the CAN controller with dedicated opto-couplers. The CAN transceiver is also galvanically isolated from the CAN controller and the Raspberry Pi in general with the same high-efficiency DC-DC converter used to power the RS-485 line drivers.

The IGND terminal is the isolated ground reference for the CAN port. Always use this line, not GND, for the CAN connection.

A 120 Ohm termination resistor between CANH and CANL can be enabled installing jumper JP3, located on the bottom side of the base board, opposite the CAN terminal block.



CAN TERMINATION JUMPER SCHEMATIC



CAN TERMINATION JUMPER POSITION

EXP port

The EXP port is a proprietary 5V bidirectional I2C interface exposed on a 6-position RJ12 socket that can be used to connect external devices.

NOTICE

All lines of the EXP port are at 5V level. Do not connect incompatible devices, like 3.3V I2C devices, telephone equipment or any other device not specifically designed to be connected to this proprietary port.

The I2C bus is driven by the SAME54 microcontroller. Iono Pi Max standard firmware implements an I2C bridge, that allows to automatically bridge and perform address translation of I2C commands with specific device addresses from the main I2C interface of the Raspberry Pi to this port (I2C control registers 17-18).

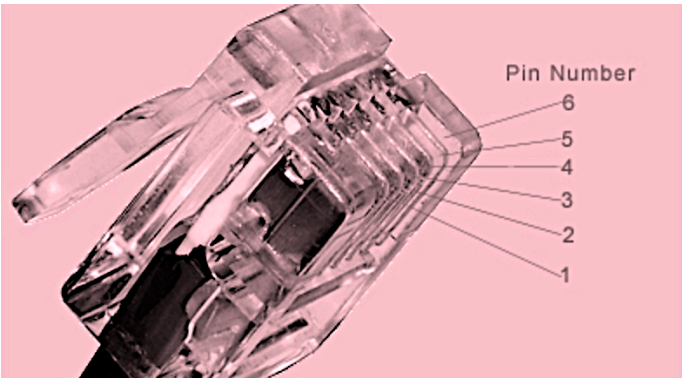
The I2C bus speed is 125 Kbps.

The EXP I2C bus driver is disabled by default, and can be enabled writing to the sysfs file expbus/enabled (I2C control register 137) exposed by the Iono Pi Max kernel module.

Besides the 5V-level SDA and SCL lines, the EXP interface also carries a 5V, 0.5A power supply line, that can be turned on or off writing to expbus/5vx. This power supply line is also protected from short-circuit or overload, reported on sys_state/5vx_err (I2C control register 140). This 5V line also powers the 5V side of the I2C driver and the lines pull-ups, so it must be enabled unless the 5V is provided by an external device.

Finally, the AUX line, normally at a high voltage level, can be driven low from any device connected to the EXP bus for application purposes. Its status can be checked on expbus/aux or sys_state/expbus_aux (I2C control register 140).

Pin #	Description
1	5 VDC out
2	AUX in/out (active low)
3	SCL
4	SDA
5	GND
6	GND



EXP INTERFACE PLUG PINS

SD matrix

Iono Pi Max implements a flexible dual microSD card architecture.

Iono Pi Max routes its two microSD card holders to the Compute Module through a high-speed switching matrix, controlled by the on-board microcontroller. Each SD card can be set as the boot card, with the other available as additional storage.

This enables for separate OS/data storage, SD redundancy and in-field full-system upgrades.

The BCM283x in the Raspberry Pi Compute Module supports two SD card interfaces, SD0 and SD1.

The first (SD0) is a proprietary Broadcom controller that does not support SDIO and is the primary interface used to boot and talk to the eMMC or SDX signals. The CM will always try to boot from this interface first.

The second interface (SD1) is standards compliant and can interface to SD, SDIO and eMMC devices.

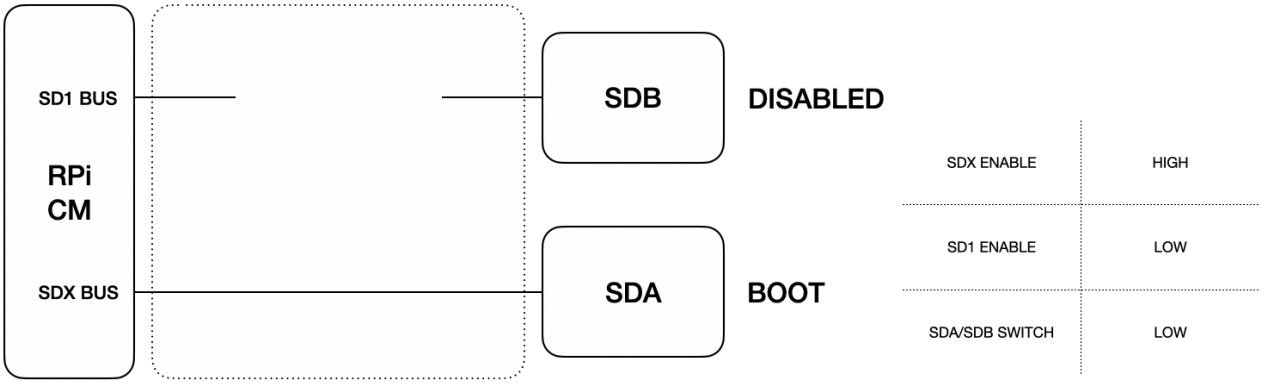
Both interfaces can support speeds up to 50MHz single ended (SD High Speed Mode), and this speed is supported by the Iono Pi Max SD matrix as well.

Iono Pi Max has two microSD slots, called SDA (bottom) and SDB (top), and the SD matrix allows 2x2 connections between both SDX/SD1 interfaces and the SDA/SDB slots.

NOTICE

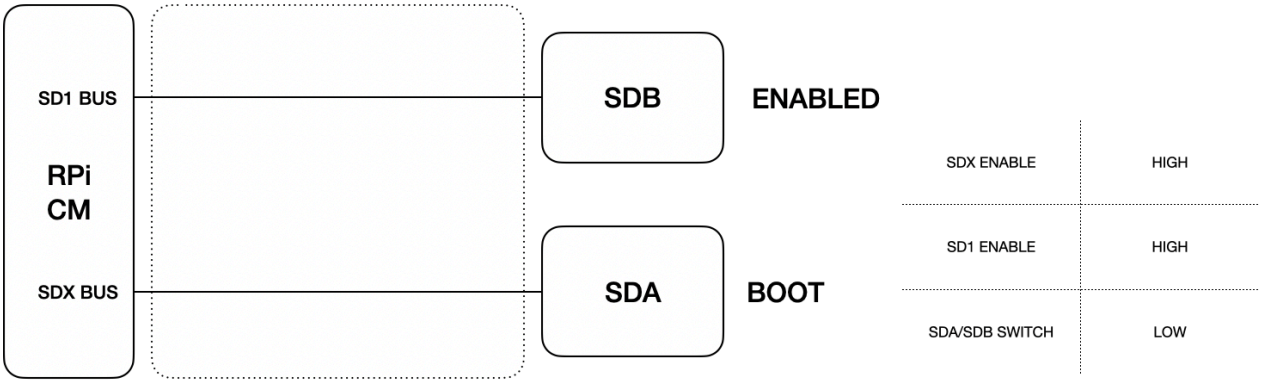
Disconnecting an SD card from the Compute Module's SD interfaces when partitions from the card are mounted at the operating system level, can result in file system and data corruption. The Iono Pi Max matrix and the microcontroller don't check for card usage and don't provide any safety mechanism to prevent improper disconnection of the SD cards.

The matrix is controlled by a few I/O lines of the Iono Pi Max SAME54 microcontroller. These lines are managed using the microcontroller I2C control registers and the Iono Pi Max kernel driver for Linux. The matrix has a hardware default configuration so that, when at power-up all microcontroller lines are high-impedance, it has a stable configuration, with SDX connected to SDA and SD1 not connected. When the microcontroller is reset, even if the control lines change state, the matrix control logic will retain the current configuration.



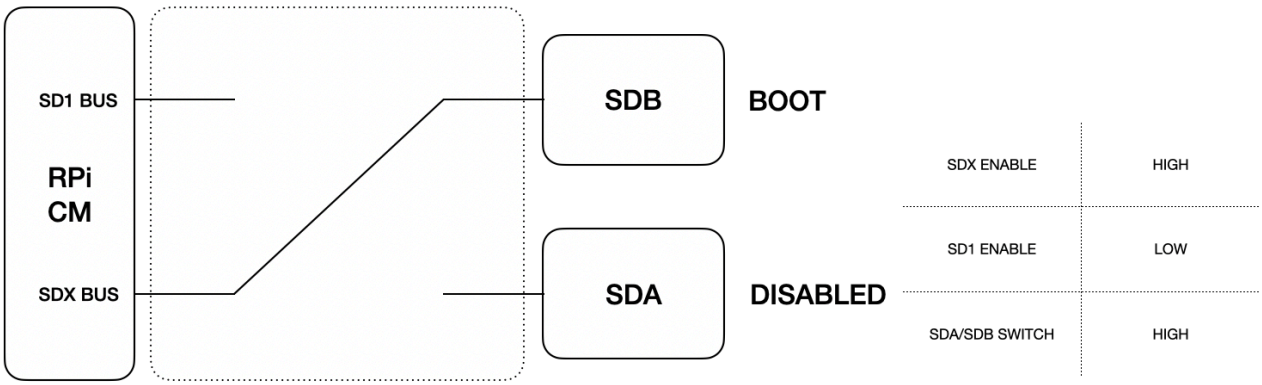
SD MATRIX DEFAULT CONFIGURATION

With SDA used as the boot card, SDB can be connected to the SD1 interface, and used as additional storage.



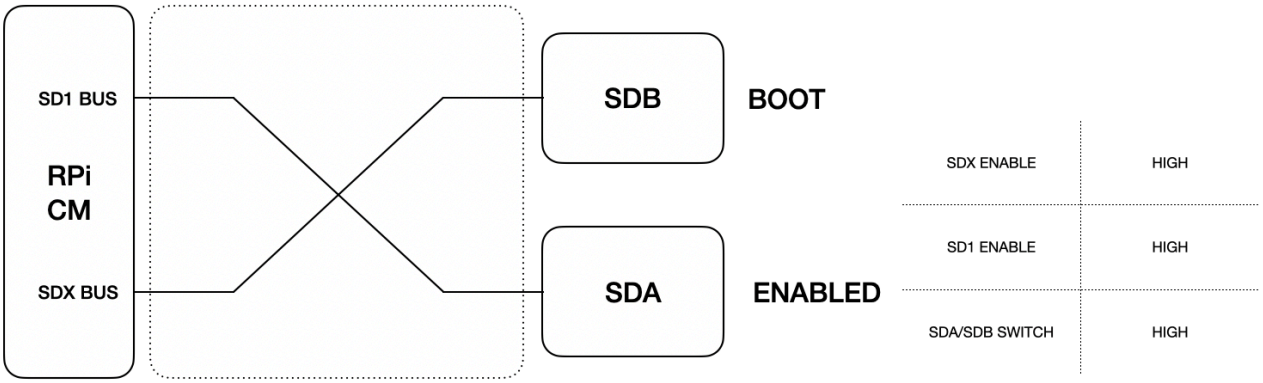
SD MATRIX BOOT FROM SDA AND SDB ENABLED FOR ADDITIONAL STORAGE

The SDX interface can be connected to SDB instead of SDA, so that boot will be done from SDB.



SD MATRIX BOOT FROM SDB

Finally, with the SDX interface connected to SDB for boot, the SDA slot can be enabled to be used for additional storage.



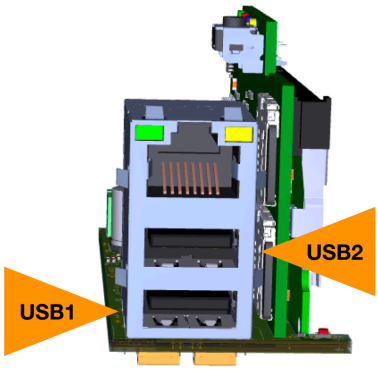
SD MATRIX BOOT FROM SDB AND SDA ENABLED FOR ADDITIONAL STORAGE

It is also possible to disable both the SDX and SD1 interfaces. This configuration could be used with the Compute Module versions that have the embedded eMMC storage, when no external SD card is needed.

The sysfs files controlling the matrix are sdx_enabled, sd1_enabled, sdx_default and sdx_routing, in the sd/ directory (I2C control register 52).

USB ports power management

Iono Pi Max has two external USB ports, USB1 (bottom) and USB2 (top), and one additional internal port.



USB PORTS

These ports are individually controlled by the MIC2076, a power distribution controller with circuit protection. The MIC2076 is internally current limited and has thermal shutdown that protects the device and load (0.5A max). A fault status output flag is asserted during over-current and thermal shutdown conditions. The top and internal ports share the same power control line, and can't be powered independently. They share the 0.5A maximum load.

The power-enable inputs and the fault status outputs are routed to the Raspberry Pi's GPIO lines to easily control and monitor connected USB devices from your application.

Power is on by default. GPIO30 and GPIO31 on the Pi control power to USB1 and USB2 respectively Set GPIO30/31 high to remove power. The sysfs files controlling these lines are usb1_enabled (bottom port) and usb2_enabled (top and internal ports), in the usb/ directory.

The USB ports are powered down during controlled power cycles (see below).

GPIO0 and GPIO1 return the fault state of USB1 and USB2. These lines will be high in normal conditions, or low when a fault is present. They are mapped to sysfs files usb1_err and usb2_err in the usb/ directory.

ATECC608 secure element

Iono Pi Max embeds the ATECC608A or ATECC608B (depending on production lot) secure element chip from Microchip. The ATECC608 is connected to the Raspberry Pi Compute Module via the I2C bus (default address: 0x60).

Its key features are:

- ✓ Cryptographic co-processor with secure hardware-based key storage
- ✓ Protected storage for up to 16 Keys, certificates or data
- ✓ Hardware support for asymmetric sign, verify, key agreement – ECDSA: FIPS186-3 Elliptic Curve Digital Signature
    - ✓ ECDH: FIPS SP800-56A Elliptic Curve Diffie-Hellman
    - ✓ NIST standard P256 elliptic curve support
- ✓ Hardware support for symmetric algorithms
    - ✓ SHA-256 & HMAC hash including off-chip context save/restore
    - ✓ AES-128: encrypt/decrypt, Galois field multiply for GCM

- ✓ Networking key management support
  - ✓ Turnkey PRF/HKDF calculation for TLS 1.2 & 1.3
  - ✓ Ephemeral key generation and key agreement in SRAM – Small message encryption with keys entirely protected
- ✓ Secure boot support
  - ✓ Full ECDSA code signature validation, optional stored digest/signature – optional communication key disablement prior to secure boot
  - ✓ Encryption/Authentication for messages to prevent on-board attacks
- ✓ Internal high-quality FIPS 800-90 A/B/C Random Number Generator (RNG)
- ✓ Two high-endurance monotonic counters
- ✓ Guaranteed unique 72-bit serial number.

Note that, while by default the ATECC608 is connected to the Pi's I2C bus, it could be re-routed, using custom firmware, to the SAME54 I2C bus (DMZ_I2C).

EERAM

Iono Pi Max has a Microchip 47L16 16 Kbit SRAM with EEPROM Backup. This chip combines the persistent storage characteristics of a traditional EEPROM, without the limitations in terms of number of erase cycles. It is connected to the SAME54 I2C bus (DMZ_I2C).

Push button

Iono Pi Max has a user-configurable push button on the front panel. When this button is pressed, the Iono Pi Max controller will set GPIO38 high, so that an application can easily detect it by simply polling this GPIO line. The button status is also available reading the sysfs button/status file.

LEDs

Iono Pi Max has five RGB LEDs visible through the front panel. At power-up, all these LEDs briefly turn on green.

An additional blue LED on the bottom and top circuit boards, that is not visible from the outside, flashes continuously when Iono Pi Max is powered and the SAME54 control firmware is running. It remains steadily on when the firmware is not running (microcontroller reset or not programmed, or critical firmware error conditions).

Of the five front panel RGB LEDs, four are assigned by default to specific functions: PWR, for power and system status, RS-232, RS-485 and CAN TX (blue) / RX (green) activity.

The USR LED is not associated to a specific function, but reserved for user control.

Besides the default functions, all LEDs can be overridden and controlled by application software.

Each LED has controllable general brightness, and independent brightness control of the primary red, green and blue colors. The sysfs control files are l<n>_br, l<n>_r, l<n>_g and l<n>_b in the led/ directory (I2C control registers 105-108, 110-113, 115-118, 120-123, 125-128 for CAN, RS-485, RS-232, USR and PWR).

When the user changes any of the control registers, that specific color or brightness level is released for exclusive user's control, and will not be driven again, until the microcontroller is reset, to provide the default indications.

Power supply

The power supply input voltage and current can be monitored reading files power_in/mon_v, in mV (I2C control register 145), and power_in/mon_i, in mA (I2C control register 146).

Note that, when Iono Pi Max is being powered by the UPS battery, power_in/mon_v reads a voltage value close to zero, while power_in/mon_i returns the current drawn from the battery.

You can start a delayed power cycle by rising the Compute Module GPIO18 pin, writing 1 to the power/down_enabled file.

The timing and behaviour of the power cycle procedure are controlled by several configuration parameters, described below.

After being powered off, Iono Pi Max will supply power to the Compute Module only when the main power source is restored (the power-up mode parameter can change this behavior, so that Iono Pi Max will always power up the Compute Module after a power cycle, even when powered by the battery).

Shutdown enable mode

Set the shutdown enable mode.

In the default mode, the power-cycle is activated rising the Compute Module GPIO18 pin.

In the alternative "arm mode", setting GPIO18 high will arm the shutdown procedure, but will not start the power-cycle until GPIO18 goes low. After the line goes low, the Iono Pi Max controller will wait 60 seconds (configurable with the shutdown wait time parameter) and then power off the Pi board for at least 5 seconds (configurable with the power-off time parameter).

Note that, if the watchdog is enabled in "arm mode", it will remain active while GPIO18 is high (armed).

Using the alternate "arm mode" a shorter shutdown wait time could be configured, then GPIO18 set high immediately before starting the OS shutdown procedure. The GPIO18 pin should go low automatically when the CPU is halted, so that the power cycle will occur shortly after. If the shutdown procedure fails to complete, the watchdog will still intervene to prevent an indefinite lock.

The shutdown enable mode is configured with the power/down_enable_mode file (I2C control register 36).

Shutdown wait time

Set the shutdown wait time, in seconds.

The factory default is 60 seconds, configurable with the power/down_delay file (I2C control register 37).

Power-off time

Set the power-off time, in seconds.

The factory default is 5 seconds, configurable with the power/off_time file (I2C control register 38).

Power-up delay

Set the power-up delay, in seconds.

After being powered off, the Iono Pi Max controller will supply power to the Compute Module only when the main power source is restored for at least the number of seconds set with this parameter, or immediately if the power-up delay is set to 0.

The factory default is 0 seconds (immediate power-up), configurable with the power/up_delay file (I2C control register 39).

Power-up mode

Set the power-up mode.

If the GPIO18 pin is set to high when the main power is not present, the Compute Module and the whole Iono Pi Max will be powered down after the shutdown wait time, and turned on again only when the main power is restored.

You can change this behavior so that only the Compute Module is turned off, and the power is always restored after the power-off time, even if running on battery, with no main power present.

The power-up mode is configured with the power/up_mode file (I2C control register 36).

UPS and battery charger

When an external lead-acid rechargeable battery is connected to Iono Pi Max battery terminals, the Raspberry Pi Compute Module and all Iono Pi Max features will remain powered when the main power supply fails, as long as there is enough energy in the battery.

When power is available, the external rechargeable battery is charged with a current that depends on the battery voltage and capacity, as configured, or kept fully charged with a minimal floating current.

Battery operations status is visually notified with the PWR LED and can be checked reading the I2C control registers or the corresponding sysfs files in the ups/ directory.

Starting from a power off condition, Iono Pi Max will not power-up if only the back-up battery is connected.

Battery voltage and charge current can be monitored reading files ups/charger_mon_v, in mV (I2C control register 147), and ups/charger_mon_i, in mA (I2C control register 148).

Note that, when Iono Pi Max is being powered by the UPS battery, ups/charger_mon_v continues to show the battery voltage, while ups/charger_mon_i returns a value close to zero.

Auxiliary voltage outputs

Iono Pi Max has an auxiliary regulated voltage output, labeled VSO on the terminal block, that can be set in a 12Vdc to 24Vdc voltage range, and is powered by the UPS.

In the factory default configuration, VSO is enabled, and set for 12V output voltage.

VSO can be enabled/disabled writing 1 or 0 to the power_out/vso_enabled sysfs file (I2C control register 132).

The output voltage, from 11.5V to 24.5V, is set writing the voltage in mV to power_out/vso (I2C control register 133).

The read-only sys_state/vso_prot (I2C control register 140) file returns 0 in normal conditions, or 1 if the VSO output is in over-current protection, with the output temporarily disabled.

The read-only sys_state/vso_err (I2C control register 140) file returns 0 in normal conditions, or 1 if the Iono Pi Max controller is unable to control the voltage regulator setting. This should never happen, and most likely indicates a hardware failure. The controller automatically disables the VSO output if this error occurs.

Iono Pi Max also has an auxiliary 5Vdc voltage output pin on the terminal block, labeled 5VO, to supply power to external 5V loads, like 1-Wire devices. The current output limit is 500mA. An internal protection circuit shuts down the 5VO line if the current limit is exceeded.

In the factory default configuration, 5VO is enabled.

5VO can be enabled/disabled writing 1 or 0 to the power_out/5vo_enabled sysfs file (I2C control register 137).

The read-only sys_state/5vo_prot (I2C control register 140) file returns 0 in normal conditions, and 1 if the 5VO output is in over-current protection, with the output temporarily disabled.

Hardware watchdog

Iono Pi Max implements a dedicated hardware watchdog circuit that can be used to perform a power cycle of the Raspberry Pi Compute Module.

Being controlled by Raspberry Pi's GPIO pins, it is extremely easy to control watchdog operation using simple custom scripts in your application.

The watchdog can also swap the boot SD card after one or more consecutive watchdog timeouts, to automatically recover from SD cards failures.

Watchdog default behavior is described below, but all timing parameters and control logic can be modified through the I2C control registers and sysfs files.

The watchdog is normally disabled. To enable it, set GPIO39 pin to high. While GPIO39 is high, the Iono Pi Max controller will watch for state changes of the GPIO32 heartbeat pin. You should ensure that GPIO32 flips its state between high and low faster than every 60 seconds.

If the Iono Pi Max controller doesn't see GPIO32 changing state for more than 60 seconds, it will initiate a shutdown procedure, rising the GPIO17 timeout pin to high, to signal that a watchdog timeout has occurred.

If possible, you should immediately initiate a software shutdown on Raspberry Pi Compute Module, and set the GPIO18 shutdown pin to high. When GPIO18 is set to high or 60 seconds after the timeout, Iono Pi Max will wait 60 seconds more, then power cycle the Raspberry Pi Compute Module (removing all power supply sources to the Compute Module, and at the same time pulling the RUN line low).

Note that Iono Pi Max will reset the Raspberry Pi Compute Module even if GPIO18 is not set high. In this case the power cycles occurs 120 seconds after the heartbeat timeout.

All of the above watchdog features can also be controlled using the sysfs files under the watchdog/ directory.

Internal fan and temperature sensors

Iono Pi Max has an internal fan to improve heat dissipation inside its DIN-rail enclosure, particularly in the area of the power regulators, where most of the heat is generated.

Two temperature sensors, located on the top and bottom side of the main circuit board, can be used to monitor the current temperature. The values, in °C/100, are available in the read-only files sys_temp/top and sys_temp/bottom (I2C control files 155-156).

In the factory default configuration, the Iono Pi Max controller automatically starts the fan when the internal temperature reaches 55 °C or power supply input power reaches 15 W, and stops when temperature falls below 50 C and power supply input power becomes less than 10W.

It is also possible to force the fan to run continuously writing 1 to fan/always_on (I2C control register 137).

The fan status is available from read-only file fan/status (I2C control register 140).

Analog inputs

Iono Pi Max has voltage, current and Pt100/Pt1000 sensors analog inputs. All analog inputs are galvanically isolated from the main power supply rail and all digital logic in Iono Pi Max.

The AGND terminals are the ground reference for the analog inputs.

All these inputs are connected to a low-noise, high-resolution 24-bit analog-to-digital converter, the Analog Devices AD4112, which is factory calibrated to achieve a high degree of accuracy.

The Iono Pi Max controller's firmware takes care of the ADC configuration and reading, returning the measured values in I2C control registers, and the corresponding sysfs files.

The analog section of Iono Pi Max is enabled by default, but can be disabled, turning off the dedicated isolated DC-DC converter, the ADC and the analog front-end stage.

The sysfs analog enable file is analog_in/enabled (I2C control register 69).

The firmware also controls a high-speed filter for the analog voltage and current inputs. The control file is analog_in/hsf (I2C control register 69). This is disabled by default. Enabling the high-speed filter results in a faster sampling rate, but also increased noise over measured values.

When all inputs are enabled the data refresh period for voltage and current inputs is 900ms, and approximately 200ms with the high-speed filter enabled. Inputs can be individually disabled (I2C control register 70) to achieve higher sample rates on enabled ones. Note that the AD4112 ADC can achieve higher SPS (samples per second) rates than the data refresh period supported by the Iono Pi Max firmware, and can be implemented with custom firmware.

AVx: Analog voltage inputs

Iono Pi Max has 4 single-ended analog voltage inputs that can be individually configured in unipolar mode, with a 0-20V range, or in bipolar mode, with a -10V to +10V range.

The ADC is factory calibrated to achieve a ±0.06% accuracy (at 25°C). The voltage inputs front-end impedance is ≥1 MOhm.

NOTICE

The voltage inputs are protected from electrostatic discharge with transient voltage suppression diodes. The AD4112 voltage inputs have a ±50V absolute maximum rating. Connecting a voltage source outside the measuring range could damage the ADC or other electronic components.

You can set the unipolar or bipolar configuration with the analog_in/av<n>_mode (I2C control register 69). Measured values in mV/100 are available in the analog_in/av<n> files (I2C control registers 71-74, 24 bits resolution).

AIx: Analog current inputs

Iono Pi Max has 4 analog current inputs that can be individually configured in unipolar mode, with a 0-20mA range, or in bipolar mode, with a -10mA to +10mA range.

The ADC is factory calibrated to achieve a ±0.08% accuracy (at 25°C). The current inputs front-end impedance is 240 Ohm.

NOTICE

The current inputs are protected from electrostatic discharge with transient voltage suppression diodes. Resettable fuses on each input line also protect from over-current events. Even with these protections in place, input currents exceeding 50mA could damage the ADC or other electronic components.

You can set the unipolar or bipolar configuration with the analog_in/ai<n>_mode (I2C control register 69). Measured values in μA are available in the analog_in/ai<n> files (I2C control registers 75-78, 24 bits resolution, signed two's complement).

ATx: Temperature sensors inputs

Iono Pi Max has two inputs dedicated to connect two-wire resistance temperature sensors (RTD). Both Pt100 and Pt1000 standard sensors are supported.

To use these inputs, you should first set the sensor type in analog_in/at<n>_mode (I2C control register 69). 0 disables, the input, 1 configures a Pt100, and 2 a Pt1000.

Measured temperature values in °C/100 are available in the analog_in/at<n> files (I2C control registers 79-80, 24 bits resolution, signed two's complement).

The data refresh period for these inputs is 1000ms.

AOx: Analog voltage and current outputs

Two analog outputs can be independently configured as 0-10V voltage or 0-20mA current drivers. The current outputs are active, as they supply, not sink, current to the current loop.

The driver front-end is based on two Texas Instruments XTR305 industrial analog voltage and current drivers, driven by the Iono Pi Max SAME54 controller's digital-to-analog converter.

Sysfs files in the analog_out directory are used to enable the outputs (ao<n>_enabled, I2C control registers 94, 99), configure the voltage/current operating mode (ao<n>_mode, I2C control registers 94, 99), set the output levels (ao<n>, I2C control registers 95, 100) and read the error status from the driver (ao<n>_err, I2C control registers 96, 101). The driver detects over-temperature, load errors and common mode errors.

The analog outputs are disabled by default.

DIx: Digital inputs

Iono Pi Max has four general purpose digital inputs. These inputs are connected to GPIO lines of the Compute Module and also to the Iono Pi Max SAME54 controller.

They are protected by a resistor network, and pulled to ground when not connected.

NOTICE

The DIx digital inputs accept positive voltage levels up to 40Vdc. Applying out of range voltages could damage the product and connected devices.

Read-only sysfs files digital_in/di<n> return the high or low status of these inputs.

OCx: Open collector outputs

Iono Pi Max has four open collector outputs, with open circuit fault detection, and over-current/short detection and protection, driven by a Maxim MAX4896 8-channel relay and load driver.

digital_out/oc<n> files (I2C control registers 89-90) are used to set the open collectors status, and read the status and error conditions.

NOTICE

The OCx outputs can operate with positive voltage levels up to 50Vdc. Applying out of range voltages could damage the product and connected devices.

NOTICE

Open circuit fault detection is implemented in the MAX4896 by a pulldown current source, along with a voltage comparator, which keeps a small current flowing when the contact is open. This might interfere with the proper functioning of the connected device. This feature can be disabled for all open collectors and relays using the digital_out/pdc sysfs file (I2C control register 137).

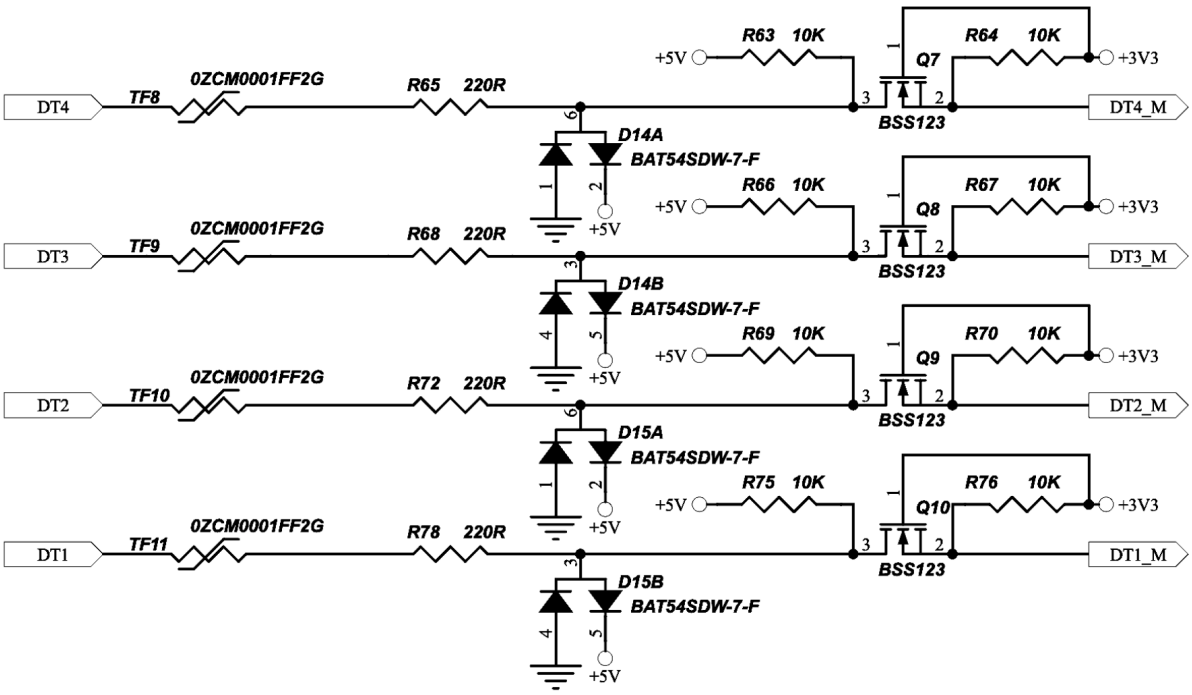
DTx: Digital input/output (Wiegand, 1-Wire)

DTx are four bidirectional TTL-level I/O lines (0÷5V), connected to the Compute Module GPIO pins through bidirectional level shifters. They can be used as general purpose bidirectional data lines, as 1-Wire bus lines, or as couples of DATA0/DATA1 lines of up to two Wiegand interfaces.

Each DTx front-end circuit has a 10.0kOhm pull-up and 220Ohm series resistors, and is protected from over-voltage and electrostatic discharge.

NOTICE

Never apply voltages outside the 0-5V range to this pin to avoid damage to the Iono Pi Max and Raspberry Pi Compute Module.



TTL LEVEL CONVERTER CIRCUIT AND PROTECTION

Ox: Relays

Iono Pi Max has four relays with change-over (CO), Single Pole Double Throw (SPDT) terminals, with normally open (NO) and normally closed (NC) contacts, driven by a Maxim MAX4896 8-channel relay and load driver.

digital_out/o<n> files (I2C control registers 84-85) are used to set the relays status, and read the status and error conditions. Note that the short circuit and open errors detect faults of the relays coils, not of the connected circuits.

Earthquake sensor module

Iono Pi Max can be shipped with an optional, factory-installed earthquake sensor module based on the OMRON D7S vibration sensor. It is connected to the SAME54 I2C bus (DMZ_I2C).

This module is soldered to the bottom side of the top circuit board (the analog, open collector and relays board) and cannot be removed by the user.

Advanced usage

Dedicated GPIO pins

Iono Pi Max uses several of the Raspberry Pi Compute Module GPIO pins. These pins should not be used for other functions.

GPIO pin	Direction	Description
GPIO2/SDA1		I²C SDA line
GPIO3/SCL1		I²C SCL line
GPIO14/TXD	out	serial TX line (RS-232 or RS-485)
GPIO15/RXD	in	serial RX line (RS-232 or RS-485)
GPIO32	out	cycle high/low for watchdog heartbeat
GPIO39	out	set to high to enable the watchdog
GPIO17	in	high on watchdog timeout
GPIO18	out	set to high to enable the shutdown cycle
GPIO40	out	set to high to turn on the buzzer
GPIO38	in	high when P1 button pressed
GPIO37	out	soft serial TX for MCU configuration console
GPIO33	in	soft serial RX for MCU configuration console
GPIO30	out	set to low to enable USB1 (bottom) power
GPIO0	in	low on USB1 fault
GPIO31	out	set to low to enable USB2 (top) power
GPIO1	in	low on USB2 fault
GPIO6	in	SDA/SDB matrix switch status
GPIO28	in	CAN controller interrupt
GPIO41	out	MCU SWD interface enable
GPIO42	in	MCU SWD interface SWO
GPIO43	in/out	MCU SWD interface SWDIO
GPIO44	out	MCU SWD interface SWCLK
GPIO45	out	MCU reset (active low)
GPIO16	in	DI1
GPIO19	in	DI2
GPIO20	in	DI3
GPIO21	in	DI4
GPIO29	in/out	DT1

GPIO pin	Direction	Description
GPIO34	in/out	DT2
GPIO35	in/out	DT3
GPIO36	in/out	DT4
GPIO8	out	SPI0 CE0 (CAN controller CS)
GPIO9	in	SPI0 MISO (CAN controller SDO)
GPIO10	out	SPI0 MOSI (CAN controller SDI)
GPIO11	out	SPI0 CLK (CAN controller SCK)
GPIO22	out	SD1 CLK
GPIO23	out	SD1 CMD
GPIO24	in/out	SD1 DAT0
GPIO25	in/out	SD1 DAT1
GPIO26	in/out	SD1 DAT2
GPIO27	in/out	SD1 DAT3

Raspberry Pi I2C bus addresses

The following table lists all components connected to the Compute Module I2C bus, and their R/W addresses.

P/N	ID	ADDR	WRITE	READ
LM75ADP	BU9	0x48 [1001000]	0x90	0x91
LM75ADP	BU38	0x49 [1001001]	0x92	0x93
MCP4652T-104E	BU7	0x2C [0101100]	0x58	0x59
MCP4562T-104E	BU42	0x2D [0101101]	0x5A	0x5B
47L16	BU39	0x50 [1010000] - SRAM	0xA0	0xA1
		0x18 [0011000] - Control Register	0x30	0x31
LP5862RSMR	VU8	0x40 [01000000]	0x80	0x81
PCF2131	VU6	0x53 [01010011]	0xA6	0xA7
ATECC608	VU3	0x60 [1100000]	0xC0	0xC1
ATSAME54P20A (firmware configuration and control registers)	BU30	0x35 [0110101]	0x6A	0x6B
D7S-A0001	TU6	0x55 [1010101]	0x99	0x9A

I2C Configuration and control registers map

The following paragraphs list the available control and configuration registers exposed by Iono Pi Max on the I2C bus, at address 0x35.

Registers containing values that span on multiple bytes store the least significant byte in the first byte (byte 0) and the most significant one in the last byte.

From firmware version 1.4 a CRC byte can optionally be used to validate the transmitted data. When reading a register of *N* bytes, an I2C read request of *N*+1 bytes will retrieve the CRC. When writing a *N*-bytes register, a write request of *N*+1 bytes implies the addition of the CRC, therefore the operation will be executed only if the CRC value is correct; if the CRC byte is not sent, the operation is always executed.

The CRC byte (8 bits) value is calculated using a CRC-8 function with polynomial $x^8 + x^5 + x^3 + x^2 + x + 1$ (i.e. 0x2F) and initial value of 0xFF. The function is applied starting with the register's address number, followed by the data bytes from least to most significant.

FIRMWARE VERSION

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	MAJOR								MINOR								R

Register 1

Bit 7-0 **MINOR**: minor version of the installed firmware

Bit 15-8 **MAJOR**: major version of the installed firmware

MCU COMMANDS

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
5	KEY								CMD								W
6					ERR	BUSY	FAIL	OK	CMD								R

Register 5

Bit 15-8 **KEY**: set to 0x2a

Bit 7-0 **CMD**: command

1 = Execute a software MCU reset

2 = Restore factory default configuration

3 = Save current configuration to be retained across power cycles

Register 6

Result of last command written to register 5

Bit 8 **OK**: set to 1 when command successfully terminated

Bit 9 **FAIL**: set to 1 when command failed

Bit 10 **FAIL**: set to 1 when command being executed

Bit 11 **FAIL**: set to 1 when invalid values are written in command register

EXP I2C BRIDGE

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
17										PI SIDE ADDRESS							R/W
18					OP CODE					DEVICE ADDRESS							R/W

Register 17

Bit 6-0 **PI SIDE ADDRESS**: I2C address mapped to the Raspberry Pi's I2C bus

In order to create or delete an entry in the I2C addresses translation table, write here the I2C address as seen on the Raspberry Pi's I2C bus, to be mapped to the EXP bus. Set register 17 before writing to register 18

Register 18

Bit 6-0 **DEVICE ADDRESS**: I2C address of the slave device

This is the I2C address of the actual device that will receive I2C commands sent to the I2C side address set in register 17. Writing register 18 commits the translation mapping configuration

Bit 11-8 **OP CODE**: translation function

0 = delete an existing translation table entry

1 = create a translation table entry from the Pi's I2C bus to the EXP port

SERIAL INTERFACES (FW VER. >= 1.6)

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
26	STP	NBIT	PAR		BAUD										TXEA	SW	R/W
27																TXE	R/W

Register 26

Bit 0 **SW**: switch RS-232/RS-485 interfaces connections

0 = RS-232 connected to UART, RS-485 connected to USB (default)

1 = RS-232 connected to USB, RS-485 connected to UART

Bit 1 **TXEA**: RS-485 TX-enable automatic control (when **SW** set to 1)

0 = TX-enable automatic control disabled

1 = TX-enable automatic control enabled, based on **STP**, **NBIT**, **PAR**, and **BAUD** parameters which must be set

Bit 11-8 **BAUD**: RS-485 baud rate for TX-enable automatic control

2 = 1200

3 = 2400

4 = 4800

5 = 9600

6 = 19200

7 = 38400

8 = 57600

9 = 115200

Bit 13-12 **PAR**: RS-485 parity for TX-enable automatic control

0 = none

1 = odd

2 = even

Bit 14 **NBIT**: RS-485 data bits for TX-enable automatic control

0 = 7 bits

1 = 8 bits

Bit 15 **STP**: RS-485 stop bits for TX-enable automatic control

0 = 1 bit

1 = 2 bits

Register 27

Bit 0 **TXE**: RS-485 TX-enable line manual control

0 = transmission disabled

1 = transmission enabled

WATCHDOG

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
29																MOD	R/W
30	TIMEOUT																R/W
31	DOWN DELAY																R/W
32	SD SWITCH																R/W

Register 29

Bit 0 **MOD**: enable mode

0 = watchdog normally disabled (default)

1 = watchdog always enabled

Register 30

Bit 15-0 **TIMEOUT**: heartbeat timeout, in seconds (default = 60)

Register 31

Bit 15-0 **DOWN DELAY**: forced power cycle delay after timeout expiration, in seconds (default = 60)

Register 32

Bit 15-0 **SD SWITCH**: SD switch control after power cycle triggered by watchdog

0 = SD switch disabled (default)

N > 0 = switch boot from SDA/SDB after N consecutive watchdog resets, if no heartbeat is detected. A value of N > 1 can be used with **MOD** set to 1 only; if **MOD** is set to 0, then **SD SWITCH** is set automatically to 1

POWER SUPPLY

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
36														SD	UM	DM	R/W
37	DOWN DELAY																R/W
38	OFF TIME																R/W
39	UP DELAY																R/W

Register 36

Bit 0 **DM**: down enable mode

0 = immediate: the power cycle is initiated as soon as GPIO18 is set to high (default)

1 = arm: setting GPIO18 high will arm the shutdown procedure, but will not start the power cycle until it goes low again (i.e. set low programmatically or Raspberry Pi switched off)

Bit 1 **UM**: up mode

0 = main power: if shutdown is enabled when the main power is not present, the Iono Pi Max is powered down after the shutdown wait time, and powered up again only when the main power is restored (default)

1 = always: if shutdown is enabled when the main power is not present, only the Raspberry Pi is turned off, and the power is always restored after the power-off time, even if running on battery, with no main power present

Bit 2 **SD**: SD switch

0 = SD switch at power cycle disabled (default)

1 = switch boot from SDA/SDB at every power cycle

Register 37

Bit 15-0 **DOWN DELAY**: shutdown delay from the moment it is enabled, in seconds (default = 60)

Register 38

Bit 15-0 **OFF TIME**: duration of power-off, in seconds (default = 5)

Register 39

Bit 15-0 **UP DELAY**: power-up delay after main power is restored, in seconds (default = 0)

UPS

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
43															BAT VOLT	EN	R/W
44	CAPACITY																R/W
45	MAX CURRENT																R/W
46	DOWN DELAY																R/W
48	CHARGE								BAT				STAT				R

Register 43

Bit 0 **EN**: UPS control

0 = disabled

1 = enabled (default)

Bit 1 **BAT VOLT**: battery voltage

0 = 12 V (default)

1 = 24 V

Register 44

Bit 15-0 **CAPACITY**: capacity of the battery connected to the UPS, in mAh.

Accepts values between 100 and 60000 (60Ah). Modifiable only when UPS is disabled (default = 800)

Register 45

Bit 15-0 **MAX CURRENT**: UPS battery maximum charging current, in mA

0 = automatically derived from capacity (default)

N > 0 = limit current below the specified value. The automatically derived limit still applies if lower than the set value

Register 46

Bit 15-0 **DOWN DELAY**: automatic power cycle timeout, in seconds.

0 = automatic power cycle disabled (default)

N > 0 = if the main power source is not available for the time set, a delayed power cycle is automatically initiated

Register 48

Bit 3-0 **STAT**: UPS status

0 = idle

1 = detecting battery

2 = battery disconnected

4 = charging battery

5 = battery charged

6 = battery in use

8 = battery overvoltage error

9 = battery undervoltage error

10 = charger damaged

11 = unstable

Bit 7 **BAT**: battery/main power

0 = running on main power

1 = running on battery power

Bit 15-8 **BAT**: estimated battery charge percentage

SD CARD SWITCH

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
52													STAT	DEF	SD1	SDX	R/W

Register 52

Bit 0 **SDX**: SDX bus control

0 = disabled

1 = enabled (default)

Bit 1 **SD1**: SD1 bus

0 = disabled

1 = enabled (default)

Bit 2 **DEF**: power-up default routing

0 = SDX bus routed to SDA and SD1 bus to SDB (default)

1 = SDX bus routed to SDB and SD1 bus to SDA

Bit 3 **STAT**: current SD routing

0 = SDX bus routed to SDA and SD1 bus to SDB

1 = SDX bus routed to SDB and SD1 bus to SDA

ANALOG INPUTS

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
69	AT2		AT1		AI4	AI3	AI2	AI1	AV4	AV3	AV2	AV1			HSF	EN	R/W
70									AI4E	AI3E	AI2E	AI1E	AV4E	AV3E	AV2E	AV1E	R/W

Register 69

Bit 0 **EN**: analog converter control

0 = disabled (power off)

1 = enabled (default)

Bit 1 **HSF**: high speed filter for AV/AI inputs

0 = disabled (default)

1 = enabled

Bit 4...7 **AVx**: AV input mode

0 = unipolar (default)

1 = bipolar

Bit 8...11 **AIx**: AI input mode

0 = unipolar (default)

1 = bipolar

Bit 13-12, 15-14 **ATx**: AT input mode

0 = disabled (default)

1 = enabled as Pt100 sensor input

2 = enabled as Pt1000 sensor input

Register 70 (FW >= 1.3)

Bit 0...3 **AVxE**: AV input control

0 = disabled

1 = enabled (default)

Bit 4...7 **AIxE**: AI input control

0 = disabled

1 = enabled (default)

ADDR	BYTE 2								BYTE 1								BYTE 0								R/W
	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
71	AV1																								R/W
72	AV2																								R/W
73	AV3																								R/W
74	AV4																								R/W
75	AI1																								R/W
76	AI2																								R/W
77	AI3																								R/W
78	AI4																								R/W
79	AT1																								R/W
80	AT2																								R/W

Registers 71...74

Bit 23-0 **AVx**: signed measured voltage value, in mV/100

Registers 75...78

Bit 23-0 **AIx**: signed measured current value, in μA

Registers 79, 78

Bit 23-0 **ATx**: signed measured temperature value, in °C/100

RELAYS

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
84					MASK								O4	O3	O2	O1	(R)/W
85									04S		03S		O2S		O1S		R

Register 84

Bit 0...3 **Ox**: relay control

0 = open (default state)

1 = closed

Bit 11...8 **MASK**: bitmap mask for Ox bits (write only)

Register 85

Bit 0-1, 3-2, **OxS**: relay status

5-4, 7-6 0 = open (default state)

1 = closed

2 = fault open

3 = short circuit

OPEN COLLECTORS

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
89					MASK								OC4	OC3	OC2	OC1	(R)/W
90									0C4S		0C3S		OC2S		OC1S		R

Register 84

Bit 0...3 **OCx**: open collector control

0 = open (default state)

1 = closed

Bit 11...8 **MASK**: bitmap mask for OCx bits (write only)

Register 85

Bit 0-1, 3-2, 5-4, 7-6 **OCxS**: open collector status

0 = open (default state)

1 = closed

2 = fault open

3 = short circuit

ANALOG OUTPUTS

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
94															I/V	EN	R/W
95	AO1																R/W
96														EFCM	EFLD	EFOT	R

Register 94

Bit 0 **EN**: AO1 control

0 = disabled (default)

1 = enabled

Bit 1 **I/V**: AO1 current/voltage mode

0 = current

1 = voltage

Register 95

Bit 15-0 **AO1**: AO1 value, in mV (voltage mode) or μA (current mode)

Register 96

Bit 0 **EFOT**: set to 1 on over-temperature error

Bit 1 **EFLD**: set to 1 on load error

Bit 2 **EFLD**: set to 1 on common mode error

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
99															I/V	EN	R/W
100	AO2																R/W
101														EFCM	EFLD	EFOT	R

Register 99

Bit 0 **EN**: AO2 control

0 = disabled (default)

1 = enabled

Bit 1 **I/V**: AO2 current/voltage mode

0 = current

1 = voltage

Register 100

Bit 15-0 **AO2**: AO2 value, in mV (voltage mode) or μA (current mode)

Register 101

Bit 0 **EFOT**: set to 1 on over-temperature error

Bit 1 **EFLD**: set to 1 on load error

Bit 2 **EFLD**: set to 1 on common mode error

LEDS

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
105									L1 RED								R/W
106									L1 GREEN								R/W
107									L1 BLUE								R/W
108									L1 BRIGHTNESS								R/W
110									L2 RED								R/W
111									L2 GREEN								R/W
112									L2 BLUE								R/W
113									L2 BRIGHTNESS								R/W
115									L3 RED								R/W
116									L3 GREEN								R/W
117									L3 BLUE								R/W
118									L3 BRIGHTNESS								R/W
120									L4 RED								R/W
121									L4 GREEN								R/W
122									L4 BLUE								R/W
123									L4 BRIGHTNESS								R/W
125									L5 RED								R/W
126									L5 GREEN								R/W
127									L5 BLUE								R/W
128									L5 BRIGHTNESS								R/W

Register 105 - 128

Bit 7-5 LED color channels (red, green, blue) and brightness values (0-255)

VSO OUTPUT

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
132																EN	R/W
133	VSO																R/W

Register 132

Bit 0 **EN**: VSO control

0 = disabled

1 = enabled (default)

Register 133

Bit 15-0 **VSO**: VSO voltage value (11500 - 24500), in mV (default = 12000)

PERIPHERALS CONTROL

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
137												PDC	EXP	5VX	5VO	FAN	R/W

Register 137

Bit 0 **FAN**: fan control

1 = keep the internal fan running even when below temperature and power thresholds

0 = fan starts automatically when the internal temperature reaches 55ºC or power supply input power reaches 15W or VSO output power reaches 7W (default)

Bit 1 **5VO**: auxiliary 5V power supply output

1 = enable (default)

0 = disable

Bit 2 **5VX**: power supply line of the EXP port

1 = enable

0 = disable (default)

Bit 3 **EXP**: EXP bus driver control

1 = enable

0 = disable (default)

Bit 4 **PDC**: digital outputs (Ox/OCx) pulldown current

1 = enable (default)

0 = disable

SYSTEM STATUS

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
137			485 ERR	232 ERR	TEMP	LED ERR	UPS ERR	ADC ERR	VSO ERR	AO2 PROT	AO1 PROT	VSO PROT	EXP AUX	5VX PROT	5VO PROT	FAN	R

Register 140

Bit 0 **FAN**: fan status

0 = inactive

1 = active

Bit 1 **5VO PROT**: 5VO output protection

0 = output OK

1 = protection enabled, output temporarily disabled

Bit 2 **5VX PROT**: 5VX output protection

0 = output OK

1 = protection enabled, output temporarily disabled

Bit 3 **EXP AUX**: EXP bus auxiliary line status

0 = EXP bus auxiliary line low

1 = EXP bus auxiliary line high

Bit 4 **VSO PROT**: VSO output protection

0 = output OK

1 = protection enabled, output temporarily disabled

Bit 5, 6 **AOx PROT**: AOx output protection

0 = output OK

1 = protection enabled, output temporarily disabled

Bit 7 **VSO ERR**: VSO control status

0 = VSO control OK

1 = VSO control failure

Bit 8 **ADC ERR**: analog converter for analog inputs status

0 = ADC OK

1 = ADC failure

Bit 9 **UPS ERR**: UPS status

0 = UPS OK

1 = UPS failure

Bit 10 **LED**: LEDs module control status

0 = module OK

1 = module failure

Bit 11 **TEMP**: system temperature probes status

0 = probes OK

1 = probes failure

Bit 12 **232 ERR**: RS-232 interface status

0 = RS-232 OK

1 = RS-232 failure

Bit 13 **485 ERR**: RS-485 interface status

0 = RS-485 OK

1 = RS-485 failure

POWER MONITORING

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
145	VIN V																R
146	VIN I																R
147	BAT V																R
148	BAT I																R
149	VSO V																R
150	VSO I																R

Register 145

Bit 15-0 **VIN V**: voltage measured on power supply input, in mV

Register 146

Bit 15-0 **VIN I**: current drain measured on power supply input, in mA

Register 147

Bit 15-0 **BAT V**: voltage measured on battery charger output, in mV

Register 148

Bit 15-0 **BAT I**: current drain measured on battery charger output, in mA

Register 149

Bit 15-0 **VSO V**: voltage measured on VSO, in mV

Register 150

Bit 15-0 **VSO I**: current drain measured on VSO, in mA

SYSTEM TEMPERATURE

ADDR	BYTE 1								BYTE 0								R/W
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
155	TOP																R
156	BTM																R

Register 155

Bit 15-0 **TOP**: temperature value from sensor on the top side of the main circuit board, in °C/100

Register 156

Bit 15-0 **BTM**: temperature value from sensor on the bottom side of the main circuit board, in °C/100

Firmware upload and MCU programming

WARNING

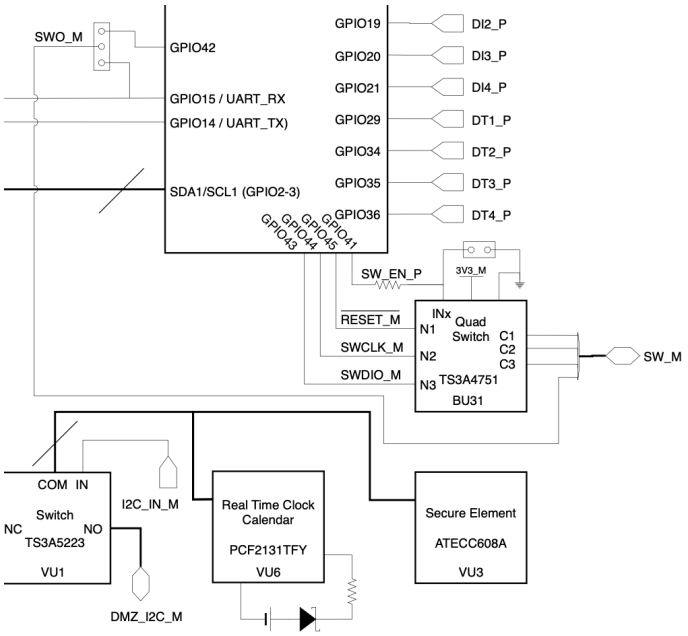
The microcontroller firmware implements critical logic and checks to ensure safe and reliable operations of Iono Pi Max including, but not limited to, fan control for heat dissipation, the battery charger logic, the VSO regulator control, watchdog, μSD switch matrix control, power distribution voltages, currents and internal temperature monitoring.

You can download the latest version of the standard firmware here:

https://www.sferalabs.cc/files/ionopimax/fw/latest/ionopimax2-fw.bin

In order to develop a safe and reliable custom firmware, you should be fully aware of the internal hardware architecture of Iono Pi Max, and of all safety and control requirements needed for your application and installation constraints.

The Iono Pi Max SAME54E microcontroller's SWD and support lines used to flash firmware and debug are connected via a high speed switch to the Raspberry Pi's GPIO41-45 pins, as you can see in the circuit block diagram detail below.



SWD LINES AND ENABLE SWITCH

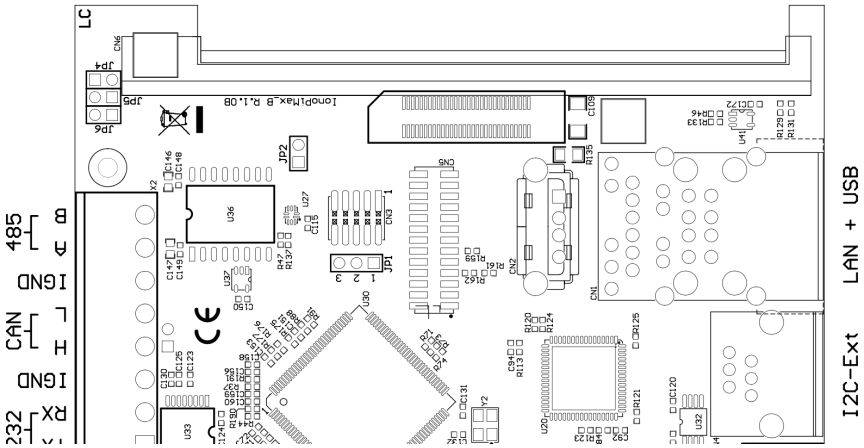
The BU31 is a high speed digital switch that prevents unintentional use of these lines, and can be used to disable, at hardware level, access to the SWD and MCU reset lines.

Installing jumper JP2 effectively disconnects these lines and prevents the Compute Module from enabling them raising GPIO41. JP2 is not factory installed, so that MCU programming is not blocked in Iono Pi Max as shipped.

If JP2 is not installed, GPIO41 must be high to enable the connection between Pi's GPIO43-45 and the MCU.

A two-way jumper, JP1, can be used if you need to feed the SWO serial debug output from the MCU to the Compute Module. Place the jumper in position 1-2 (factory installed) to route SWO to GPIO42, or in position 2-3 to route SWO to the Pi's main UART RX line (GPIO15). In this latter configuration the external serial ports must not receive traffic or be router to the MCU, not to the Compute Module.

Jumpers JP1 and JP2 are located on the main circuit board. In order to easily access them, you will need to disassemble the vertical board that hosts the Compute Module. Follow the same procedure as described to replace the real-time-clock battery.



SWD JUMPERS

One of the tools you can use on Raspberry Pi OS or other Linux distributions to flash new versions of the firmware to the MCU, or perform in-circuit debugging, is OpenOCD[6].

Install OpenOCD and its dependencies:

```
$ sudo apt install git autoconf libtool libusb-1.0-0-dev libjim-dev libgpiod-dev
$ git clone --depth 1 https://git.code.sf.net/p/openocd/code openocd
$ cd openocd
$ ./bootstrap
$ ./configure --enable-linuxgpiod --enable-bcm2835gpio
$ make
$ sudo make install
```

Create a file called "ionopimax.cfg" in /usr/local/share/openocd/scripts/interface/ with the following content:

```
adapter driver linuxgpiod
adapter gpio swclk -chip 0 44
adapter gpio swdio -chip 0 43
adapter gpio srst -chip 0 45
```

[6] http://openocd.org

Create a program file (e.g. "openocd-iono.prog") with the following content:

```
source [find interface/ionopimax.cfg]
transport select swd
source [find target/atsame5x.cfg]
reset_config srst_only
init
targets
reset halt
program your_fw_file.bin verify
reset
shutdown
```

where "your_fw_file.bin" is the path of the firmware binary file to be uploaded.

If the Iono Pi Max kernel module is enabled, disable it:

```
$ sudo rmmod ionopimax
```

Enable the SWD switch through GPIO41:

```
$ pinctrl set 41 pn op dh
```

You can now flash the MCU through the SWD interface by running the script:

```
$ sudo openocd -f openocd-iono.prog
```

If it fails, retry.

Reboot to reload the kernel module:

```
$ sudo reboot
```

Thermal considerations for Iono Pi Max

The Raspberry Pi Compute Module draws power and generates heat proportionally to CPU and GPU load.

The Iono Pi Max main regulator, battery charger and VSO regulator also generate a substantial heat proportionally to the total amount of current that has to be supplied to the Pi and, if present, external devices.

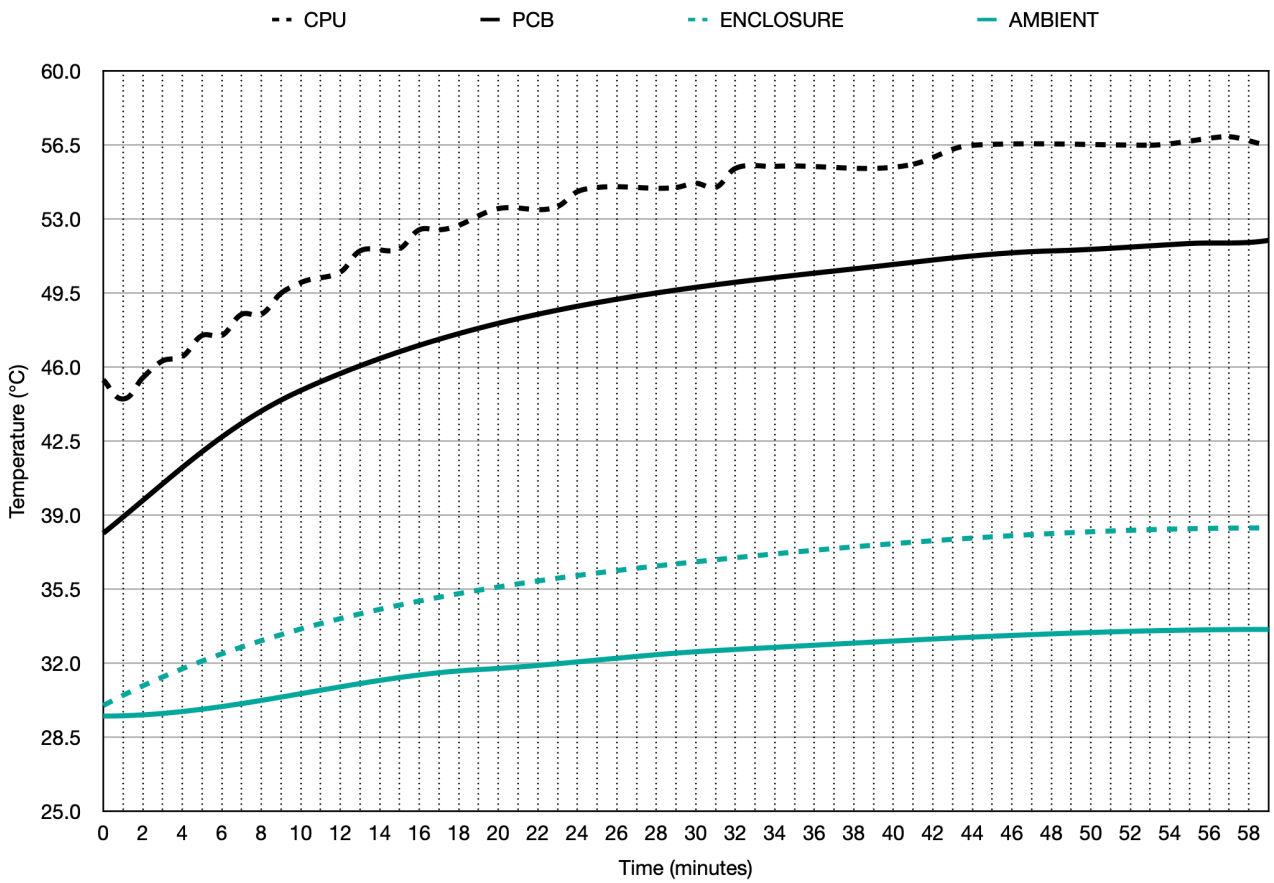
The thermal behaviour of Iono Pi Max is dynamic and affected by operating conditions and the external environment temperature.

The Raspberry Pi CPU has the ability to throttle down its clock speed once a certain core temperature is reached. This prevents CPU overheating, also limiting the temperature inside the enclosure, when the CPU load is high for a significant amount of time.

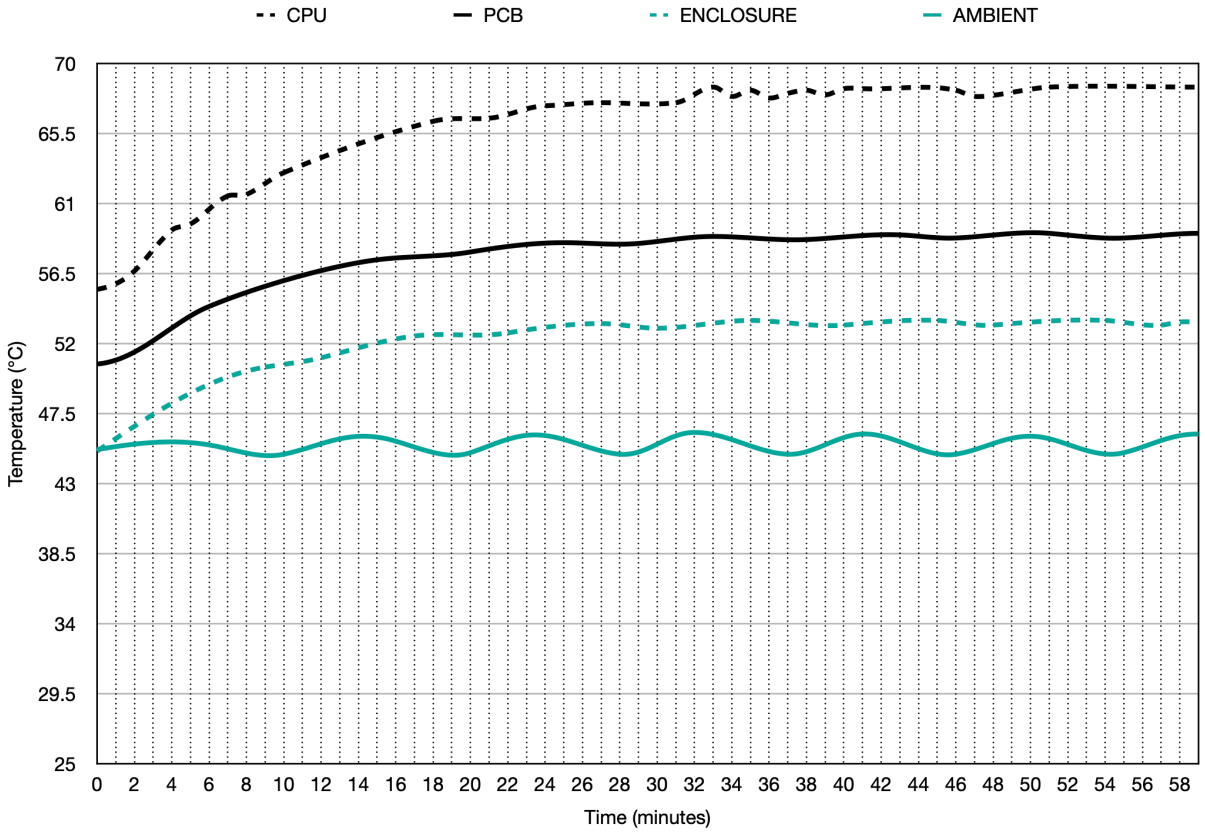
A small internal fan, directly overhead the voltage regulators area of the main circuit board, plays an important role keeping the temperature inside the enclosure to reasonable levels, even under heavy electrical load conditions. The fan starts only when temperature or power draw exceeds preset levels. See the "Internal fan and temperature sensors" chapter above for additional details.

The following charts show the thermal behaviour of Iono Pi Max in different conditions, in a typical installation with the device under test installed in a standard, not ventilated DIN cabinet:

- 30C and 45C ambient temperature (the ambient temperature is measured inside the DIN cabinet, so it is the actual air temperature surrounding the Iono Pi Max enclosure)
- Raspberry Pi Compute Module 3 B+
- Low electrical load versus high electrical load conditions
- Expect similar temperatures when using the Raspberry Pi Compute Module 4S at the same application workload, and higher temperatures under high CPU load.

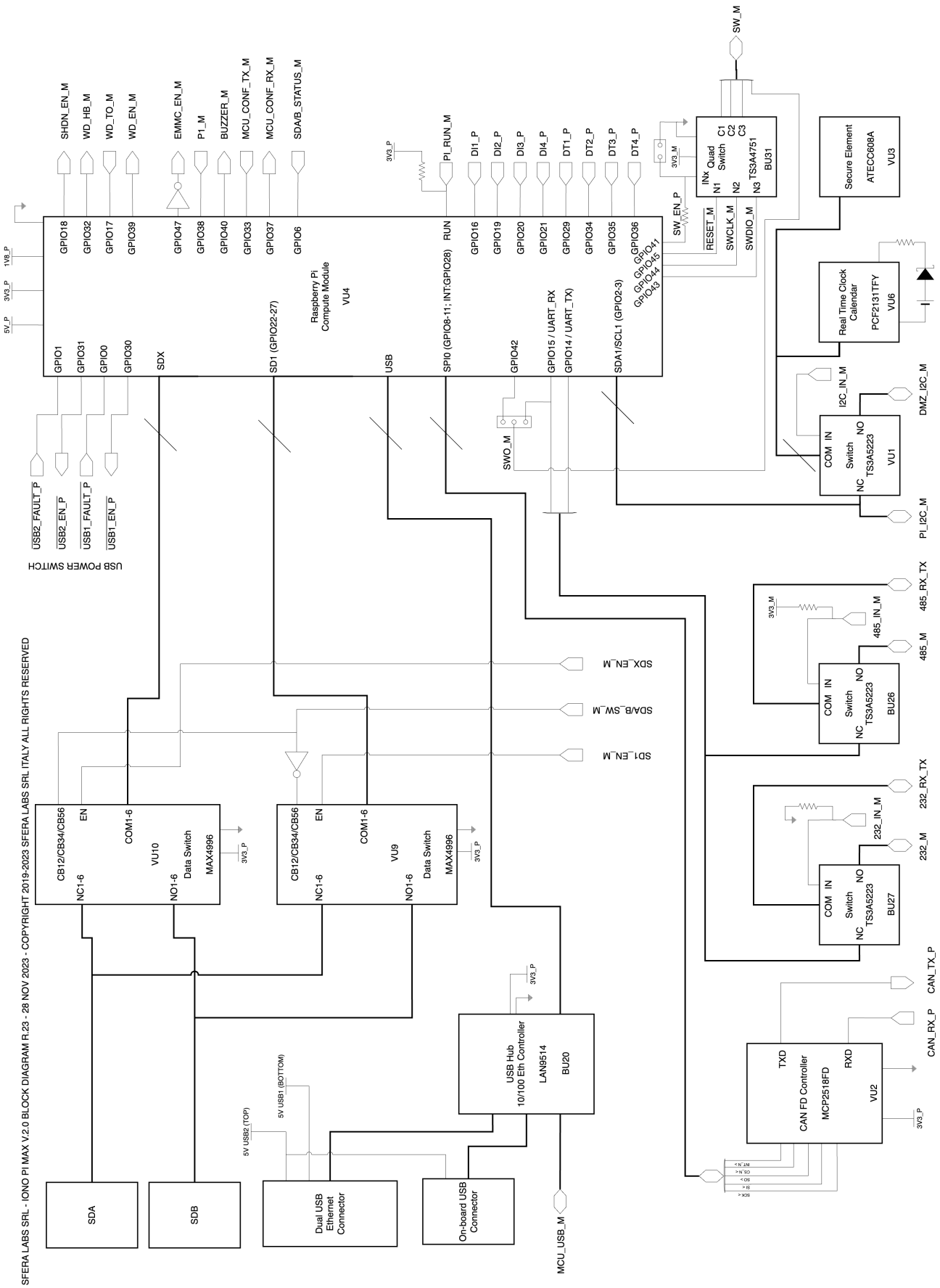


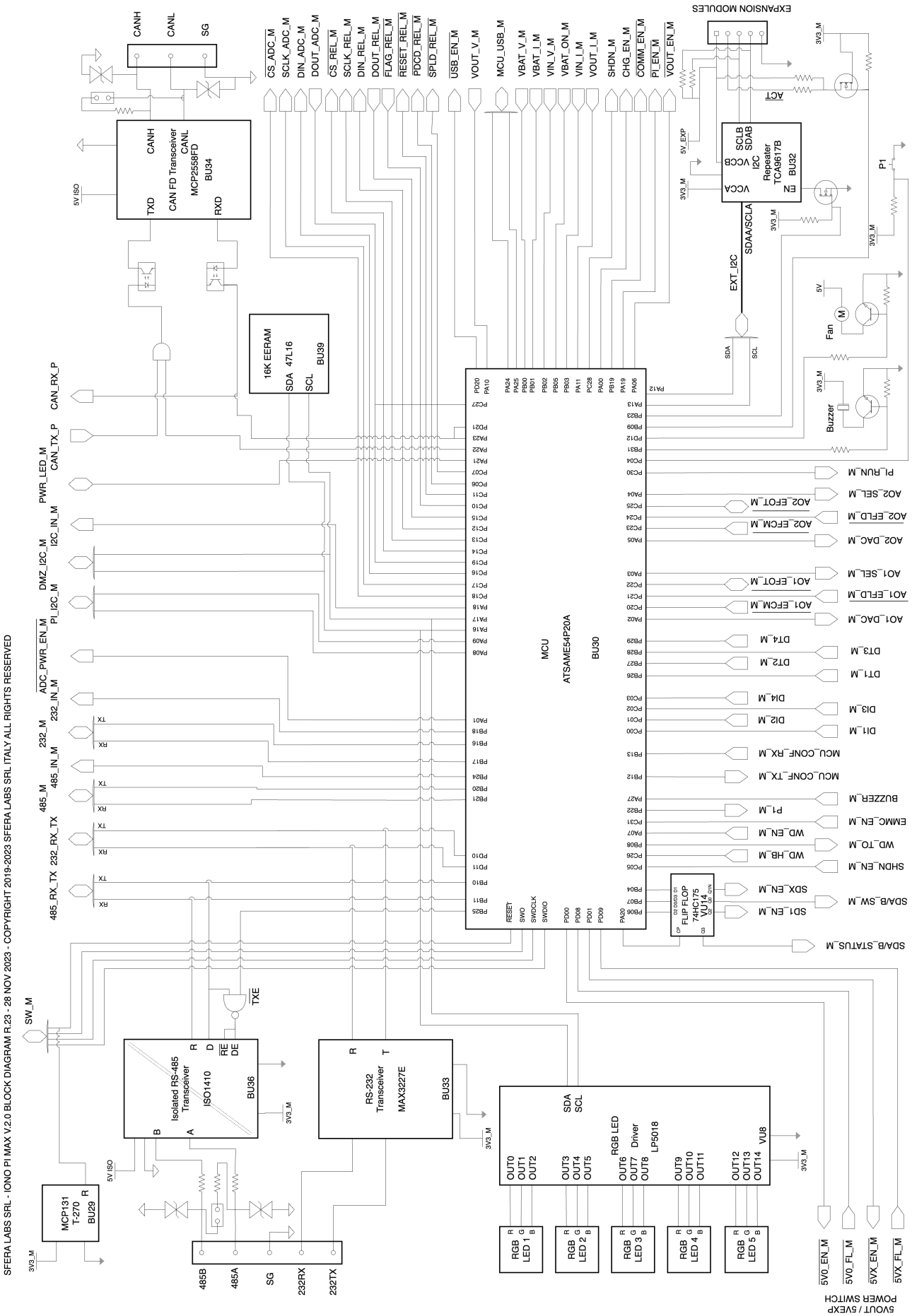
30C AMBIENT TEMPERATURE, LOW ELECTRICAL LOAD (VIN 24 V, IIN 0.15 A)

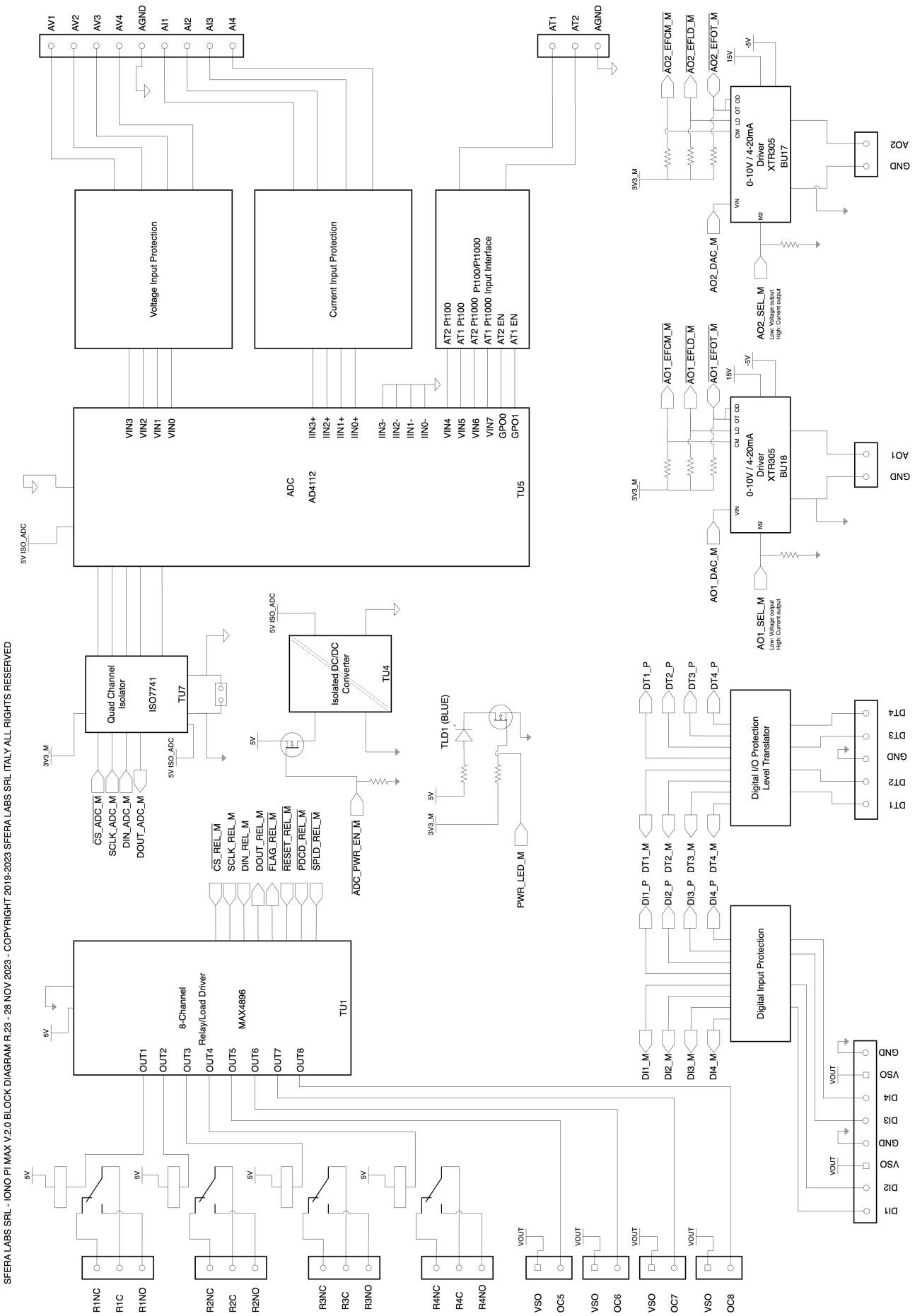


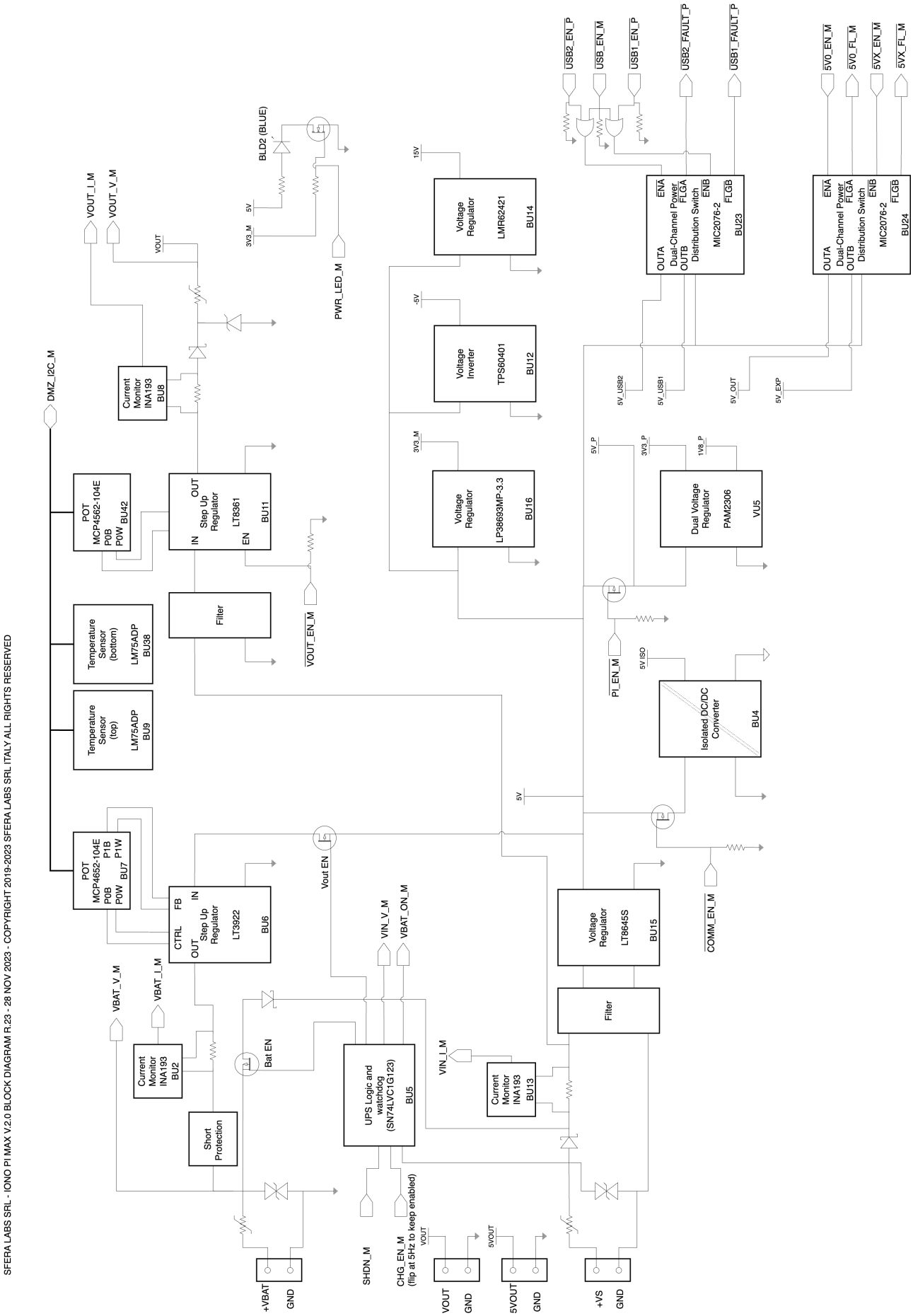
45C AMBIENT TEMPERATURE, HIGH ELECTRICAL LOAD (VIN 24 V, IIN 0.75 A)

Block diagram




SFERA LABS SRL - IONO PI MAX V.2.0 BLOCK DIAGRAM R.23 - 28 NOV 2023 - COPYRIGHT 2019-2023 SFERA LABS SRL ITALY ALL RIGHTS RESERVED
CAN FD Transceiver
MCP2558FD
BU34
16K EERAM
47L16
BU39
MCU
ATSAME54P20A
BU30
I2C Repeater
TCA9617B
BU32
FLIP FLOP
74HC175
VU14
Isolated RS-485 Transceiver
ISO1410
BU36
RS-232 Transceiver
MAX3227E
BU33
RGB LED Driver
LP5018
VU8
MCP131
T-270
BU29
EXPANSION MODULES
5V0_EN_M
5V0_FL_M
5VX_EN_M
5VX_FL_M
5VOUT / 5VEXP POWER SWITCH


SFERA LABS SRL - IONO PI MAX V.2.0 BLOCK DIAGRAM R.23 - 28 NOV 2023 - COPYRIGHT 2019-2023 SFERA LABS SRL ITALY ALL RIGHTS RESERVED
AV1
AV2
AV3
AV4
AGND
AI1
AI2
AI3
AI4
AT1
AT2
AGND
Voltage Input Protection
Current Input Protection
AT2 Pt100
AT1 Pt100
AT2 Pt1000
AT1 Pt1000
Pt100/Pt1000 Input Interface
AT2 EN
AT1 EN
VIN3
VIN2
VIN1
VIN0
IIN3+
IIN2+
IIN1+
IIN0+
IIN3-
IIN2-
IIN1-
IIN0-
VIN4
VIN5
VIN6
VIN7
GPO0
GPO1
ADC
AD4112
TU5
5V ISO_ADC
Quad Channel Isolator
ISO7741
TU7
3V3_M
5V ISO_ADC
CS_ADC_M
SCLK_ADC_M
DIN_ADC_M
DOUT_ADC_M
Isolated DC/DC Converter
TU4
5V
ADC_PWR_EN_M
TLD1 (BLUE)
PWR_LED_M
CS_REL_M
SCLK_REL_M
DIN_REL_M
DOUT_REL_M
FLAG_REL_M
RESET_REL_M
PDCD_REL_M
SPLD_REL_M
8-Channel
Relay/Load Driver
MAX4896
TU1
OUT1
OUT2
OUT3
OUT4
OUT5
OUT6
OUT7
OUT8
R1NC
R1C
R1NO
R2NC
R2C
R2NO
R3NC
R3C
R3NO
R4NC
R4C
R4NO
VSO
OC5
OC6
OC7
OC8
VOUT
AO2_EFCM_M
AO2_EFLD_M
AO2_EFOT_M
15V
-5V
0-10V / 4-20mA
Driver
XTR305
BU17
AO2
GND
3V3_M
AO2_DAC_M
AO2_SEL_M
Low: Voltage output
High: Current output
AO1_EFCM_M
AO1_EFLD_M
AO1_EFOT_M
0-10V / 4-20mA
Driver
XTR305
BU18
AO1
AO1_DAC_M
AO1_SEL_M
DT1_P
DT2_P
DT3_P
DT4_P
Digital I/O Protection
Level Translator
DT1
DT2
DT3
DT4
DI1_P
DI2_P
DI3_P
DI4_P
DT1_M
DT2_M
DT3_M
DT4_M
Digital Input Protection
DI1_M
DI2_M
DI3_M
DI4_M
DI1
DI2
DI3
DI4


SFERA LABS SRL - IONO PI MAX V.2.0 BLOCK DIAGRAM R.23 - 28 NOV 2023 - COPYRIGHT 2019-2023 SFERA LABS SRL ITALY ALL RIGHTS RESERVED
DMZ_I2C_M
POT MCP4562-104E P0B P0W BU42
Temperature Sensor (bottom) LM75ADP BU38
Temperature Sensor (top) LM75ADP BU9
POT MCP4652-104E P0B P0W P1B P1W BU7
VOUT_I_M
VOUT_V_M
VOUT
Current Monitor INA193 BU8
Step Up Regulator LT8361 BU11
IN OUT EN
Filter
VOUT_EN_M
BLD2 (BLUE)
5V
3V3_M
PWR_LED_M
Step Up Regulator LT3922 BU6
CTRL FB IN OUT
Vout EN
VBAT_V_M
VBAT_I_M
Current Monitor INA193 BU2
Short Protection
Bat EN
UPS Logic and watchdog (SN74LVC1G123) BU5
VIN_V_M
VBAT_ON_M
VIN_I_M
Current Monitor INA193 BU13
SHDN_M
CHG_EN_M (flip at 5Hz to keep enabled)
+VBAT
GND
VOUT
GND
5VOUT
GND
+VS
GND
15V
Voltage Regulator LMR62421 BU14
-5V
Voltage Inverter TPS60401 BU12
3V3_M
Voltage Regulator LP38693MP-3.3 BU16
5V
Voltage Regulator LT8645S BU15
Filter
USB2_EN_P
USB_EN_M
USB1_EN_P
USB2_FAULT_P
USB1_FAULT_P
OUTA OUTB Dual-Channel Power Distribution Switch MIC2076-2 ENA ENB FLGA FLGB BU23
5V_USB2
5V_USB1
5V_OUT
5V_EXP
5V0_EN_M
5V0_FL_M
5VX_EN_M
5VX_FL_M
OUTA OUTB Dual-Channel Power Distribution Switch MIC2076-2 ENA ENB FLGA FLGB BU24
5V_P
3V3_P
1V8_P
Dual Voltage Regulator PAM2306 VU5
PI_EN_M
5V ISO
Isolated DC/DC Converter BU4
COMM_EN_M

Technical specifications

Note: all values typical, at +25 °C and under normal operating conditions.

POWER SUPPLY	
Power supply operating voltage (VS)	⎓dc 24 V nominal (range 10-50 V), max 3.0 A Reverse polarity and surge protection with 3.3 A resettable fuse
Current consumption at VS+ 12 V⎓ including Raspberry Pi CM3+, with Ethernet connected, no USB devices, VSO off, 5V0 off, UPS enabled, battery fully charged, relays off	low CPU load: 0.31 A, 100% CPU load, before throttling: 510 mA
Current consumption at VS+ 12 V⎓ including Raspberry Pi CM3+, with Ethernet connected, no USB devices, VSO on 24 V 8 W, 5V0 on 500 mA, UPS enabled, battery fully charged, relays off	low CPU load: 1.40 A
Current consumption at VS+ 12 V⎓ including Raspberry Pi CM3+, with Ethernet connected, no USB devices, VSO on 24 V 8 W, 5V0 on 500 mA, UPS enabled, battery charging, relays all on	100% CPU load, before throttling: 1.95 A
Current consumption at VS+ 24 V⎓ including Raspberry Pi CM3+, with Ethernet connected, no USB devices, VSO off, 5V0 off, UPS enabled, battery fully charged, relays off	low CPU load: 0.17 A, 100% CPU load, before throttling: 0.27 A
Current consumption at VS+ 24 V⎓ including Raspberry Pi CM3+, with Ethernet connected, no USB devices, VSO on 24 V 8 W, 5V0 on 500 mA, UPS enabled, battery fully charged, relays off	low CPU load: 0.67 A
Current consumption at VS+ 24 V⎓ including Raspberry Pi CM3+, with Ethernet connected, no USB devices, VSO on 24 V 8 W, 5V0 on 500 mA, UPS enabled, battery charging, relays all on	100% CPU load, before throttling: 0.90 A
UPS external battery (not included)	12 V or 24 V lead acid battery. Suggested capacity: 1.2 Ah to 7 Ah. Reverse polarity and surge protection with 3.3 A resettable fuse
UPS battery charge voltage	28.5 V (Max. for 24V batteries)
UPS battery charge current	500 mA (Max. for 12V batteries)
Voltage threshold for switching to battery mode	below 7.0 V
VSO output voltage range	+12…+24 V
VSO output power	8W (Max)

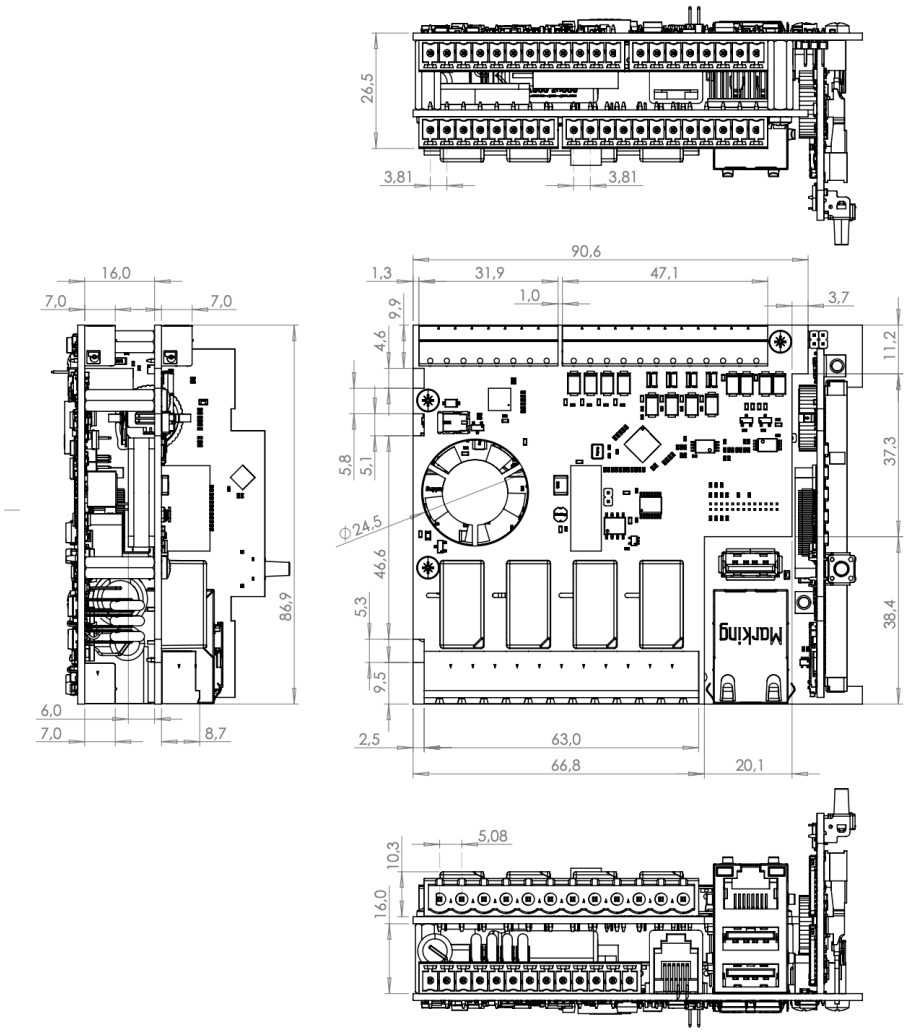
VSO output resettable fuse	1.1 A
5VO output voltage	4.7 V (no load) 4.6 V (10 Ohm load)
5VO output current	500 mA (Max)
USB1 port output current	500 mA (Max)
USB2 + USB_INTERNAL port output current	500 mA (Max)
EXP port output voltage	5 V
EXP port output current	500 mA (Max)
RASPBERRY PI COMPATIBILITY	
Raspberry platform compatibility	Raspberry Pi Compute Module 3 Raspberry Pi Compute Module 3+ Raspberry Pi Compute Module 4S
COMMUNICATION PORTS	
Serial communication ports	RS-485 Half-Duplex with automatic data direction management RS-232 Full-Duplex
Baud Rates on COMM ports	1200 to 115200
ESD-Protection Voltage on RS-232 TX/RX	±15 kV human body model ±8 kV contact discharge
ESD-Protection Voltage on RS-485 A/B	±15 kV human body model ±8 kV contact discharge
Surge protection on RS-485 A/B	Surge protection up to ±500 V / 2 Ohms 1.2/50 µs; 600 W peak pulse power capability at 10/1000 µs waveform
Fail safe feature on RS-485	Yes
CAN interface	8 Mbps (Max. CAN-FD)
ESD-Protection Voltage on CANH, CANL	±14 kV human body model ±8 kV contact discharge
Transient Voltage protection on CANH, CANL	-150 V to +100 V (waveforms of the applied transients in accordance with ISO-7637, Part 1, test pulses 1, 2, 3a and 3b)
USB ports	USB 2.0 Type A
Ethernet Port	10/100 Mbps
AVx: ANALOG VOLTAGE INPUTS	
Input voltage range	0…20 V
Input impedance	> 1 MOhm
Total Unadjusted Error (TUE)	±0.06 % of full-scale
Resolution	24 bits (Max)

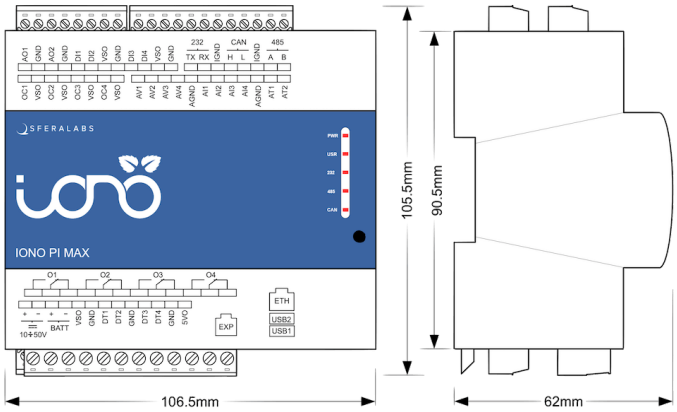
AIx: ANALOG CURRENT INPUTS	
Input current range	-0.5…+24 mA
Input impedance	240 Ohm
Total Unadjusted Error (TUE)	±0.08 % of full-scale
Resolution	24 bits (Max)
ATx: TEMPERATURE SENSORS INPUTS (PT100, PT1000)	
Temperature range	-150…+400 °C
Accuracy (not including Pt and cable errors)	± 0.3 °C ± 0.05 % of reading
Resolution	24 bits (Max)
AOx: ANALOG VOLTAGE OUTPUTS	
Output range	0…+10 V
Output drive current	15 mA (Max)
Total Error	±0.5 % of full-scale (Max)
Resolution	12 bits
AOx: ANALOG CURRENT OUTPUTS	
Output range	0…+20 mA
Output drive voltage	12 V (Max)
Total Error	±0.5 % of full-scale (Max)
Resolution	12 bits
DIx: DIGITAL INPUTS	
Input voltage range	0…40 V
Input impedance	6.9 kOhm
Voltage threshold	VIH: 4.2 V VIL: 3.4 V
OCx: OPEN COLLECTOR OUTPUTS	
Sink current	200 mA (Max)
On resistance	3 Ohm
Voltage range	0…50 V
DTx: DIGITAL INPUT/OUTPUT	
Output voltage	VOL: 0.2 V VOH: 5.0 V
Input voltage threshold	VIH: 2.4 V VIL: 0.8 V
Pull-up (+5 V)	10 kOhm

Ox: RELAYS	
Rated current	6 A
Rated voltage	250 V AC
Rated load (AC1, resistive load)	1500 VA
Rated load (AC15, cos ϕ = 0.5)	250 VA
Single phase motor rating (230 V AC)	185 W
Breaking capacity (DC1)	• 3 A at 30 V DC • 0.35 A at 110 V DC • 0.2 A at 220 V DC
Electrical life at rated load AC1	50000 cycles
Mechanical life	20000000 cycles
Operate/release time	6/4 ms
Dielectric strength between open contacts	1000 V AC
Type of insulation between coil and contacts	Basic
Dielectric strength between coil and contacts	4000 V AC
REAL TIME CLOCK	
Oscillator frequency	32768 kHz
Frequency Tolerance	±3 ppm
Backup battery	Only use CR1220 or BR1220 Lithium / Manganese Dioxide (Li/MnO2) batteries. Expected battery life without main power supply: 3 or more years, depending on environmental conditions
EERAM	
Capacity	16 Kbit (2048 x 8 bits)
SRAM r/w cycles	Infinite
EEPROM store cycles	> 1000000
Data retention	> 200 years
INTERNAL TEMPERATURE SENSORS	
Temperature accuracy	±2 °C
Resolution	11 bits (0.125 °C)
INTERNAL VOLTAGE SENSORS (VS, BATT, VSO)	
Voltage range	0…58 V (VS) 0…30 V (BATT, VSO)
Total Unadjusted Error (TUE)	±0.25 % of full-scale
Resolution	12 bits

INTERNAL CURRENT SENSORS (VS, BATT, VSO)	
Current range	0…4 A (VS) 0…1 A (BATT, VSO)
Total Unadjusted Error (TUE)	±2.5 % of full-scale
Resolution	12 bits
EMI IMMUNITY STANDARDS	
Electromagnetic immunity compliance	EN 61000-4-2 (ESD) EN 61000-4-3 (Radiated RF Field) EN 61000-4-4 (Burst/fast transient) EN 61000-4-5 (Surge) EN 61000-4-6 (Conducted) EN 61000-4-8 (Power frequency magnetic field)
ENVIRONMENTAL	
Operating temperature	-20…+60 °C
Storage temperature	-30…+80 °C
Altitude	Up to 2000 m
Humidity	5% to 95% RH noncondensing
Protection degree	IP20
MECHANICAL	
3.81 mm pitch terminal block characteristics	Maximum conductor cross section: 1.3 mm2 (16AWG), or 0.5 mm2 when using ferrules (highly recommended) Recommended stripping length: 6 mm Screw thread: M2 Maximum screws tightening torque: 0.3 Nm
5.08 mm pitch terminal block characteristics (relays)	Maximum conductor cross section: 3.3 mm2 (12AWG) Recommended stripping length: 7 mm Screw thread: M3 Maximum screws tightening torque: 0.5 Nm
Dimensions	6 module Din Rail enclosure width: 106.5 mm height: 90.5mm depth: 62.0mm
Weight	330 g (including Raspberry Pi Compute Module)

Dimensions

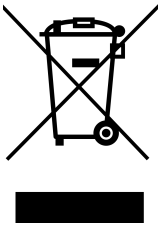




DIMENSIONS (mm)

Disposal

Waste Electrical & Electronic Equipment



(Applicable in the European Union and other European countries with separate collection systems). This marking on the product, accessories or literature indicates that the product should not be disposed of with other household waste at the end of their working life. To prevent possible harm to the environment or human health from uncontrolled waste disposal, separate these items from other types of waste and recycle them responsibly to promote the sustainable reuse of material resources. Household users should contact either the retailer where they purchased this product, or their local government office, for details of where and how they can take these items for environmentally safe recycling. This product and its electronic accessories should not be mixed with other commercial wastes for disposal.

Iono Pi Max contains a small non rechargeable manganese dioxide lithium coin battery.

In the Iono Pi Max, the battery is not accessible from the outside. You should first remove the case body to gain access to the Iono Pi Max circuit boards. Always remove the battery before disposing of this product.

Installation and use restrictions

Standards and regulations

The design and the setting up of electrical systems must be performed according to the relevant standards, guidelines, specifications and regulations of the relevant country. The installation, configuration and programming of the devices must be carried out by trained personnel.

The installation and wiring of connected devices must be performed according to the recommendations of the manufacturers (reported on the specific data sheet of the product) and according to the applicable standards.

All the relevant safety regulations, e.g. accident prevention regulations, law on technical work equipment, must also be observed.

Safety instructions

Carefully read the safety information section at the beginning of this document.

Set-up

For the first installation of the device proceed according to the following procedure:

- ✓ make sure all power supplies are disconnected
- ✓ install and wire the device according to the schematic diagrams on the specific product user guide
- ✓ after completing the previous steps, switch on the power supply and other related circuits.

Conformity Information

EU

This device complies with the following applicable European Community harmonised standards:

- ✓ 2014/30/EU - Electromagnetic Compatibility Directive (EMC)
- ✓ 2014/35/EU - Low Voltage Directive (LVD)
- ✓ 2011/65/EU and 2015/863/EU - Restriction of the use of certain hazardous substances in electrical and electronic equipment (RoHS)

The following harmonised standards have been used to demonstrate conformity to these directives:

- ✓ EN60664-1:2007 - Insulation coordination for equipment within low-voltage systems
- ✓ EN61000-6-2:2019 - EMC Immunity standard for industrial environments
- ✓ EN 61000-6-3:2007/A1:2011/AC:2012 - EMC Emission standard for residential, commercial and light-industrial environments

The declaration of conformity is available at: https://www.sferalabs.cc

USA

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna
- Increase the separation between the equipment and receiver
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected
- Consult the dealer or an experienced radio/TV technician for help.

Shielded cables must be used with this equipment to maintain compliance with FCC regulations.

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

CANADA

This Class B digital apparatus complies with Canadian ICES-003.

Cet appareil numérique de la classe B est conforme à la norme NMB-003 du Canada.

RCM AUSTRALIA / NEW ZEALAND

This product meets the requirements of the standard EN 61000-6-3:2007/A1:2011/AC:2012 - Emission for residential, commercial and light-industrial environments.