RTC MicroSD Shield

1 DESCRIPTION

The PTSolns *RTC MicroSD Shield* is intended for data logging where precise time/date stamps and/or non-volatile memory are required. The shield is outfitted with a real-time clock (RTC) and optional backup battery, a Micro SD card reader, extra EEPROM, an input/output signal port, and a Qwiic® connector.

Several hardware configuration options make this shield very flexible by allowing pin definitions to be changed by the user. The shield is fully stackable and compatible with all the common 5V Uno variations.

The *RTC MicroSD Shield* is ideal for quick prototyping, testing, and educational settings. The Qwiic connector allows for many different I2C modules to be added easily and getting a project started quickly. Several fully detailed example sketches are available to showcase all the features of the shield.

Table of Contents

2 DOCUMENT REVISION HISTORY

Current document revision is Rev 0.

3 PRODUCT FEATURES

This section highlights notable features of the *RTC MicroSD Shield.*

3.1 Real-Time Clock (RTC)

Onboard the *RTC MicroSD Shield* is the DS3231 Real-Time Clock (RTC). Of relevance to the *RTC MicroSD Shield*, the RTC can keep time from seconds to years, including leap-year compensation, up until the year 2100 with an accuracy of ±2ppm from 0°C to +40°C and ±3.5ppm from -40°C to +85°C. Furthermore, it has two alarms that can be used as interrupts, and programmable square-wave output signals.

The RTC communicates to the microcontroller development board (such as an Uno) via the I2C protocol. The I2C address is:

0x68

The RTC has a reset pin which is, by default, connected to the RST of the microcontroller development board. Resetting the RTC does not clear the time, but only the internal registers. A closed jumper pad on the back of the shield, as shown in Figure 5, can be cut to isolate the RTC reset pin from the microcontroller development board. Furthermore, the RTC outputs a software configured signal (SQW/INT), which is connected to Pin D3 via the default hardware configurations. This pin can manually be changed to Pin D2 by cutting and soldering the appropriate solder jumper on the back of the board. These hardware configuration options are discussed in Section 3.6.

Besides the SQW/INT signal, the onboard RTC also outputs pre-defined square wave output. All output signals are discussed in detailed in Section 3.7.

3.1.1 RTC Backup Battery

The user has the option to insert a CR2032 or a ML2032 coin battery into the battery holder onboard the shield, as seen in Figure 1. This battery supplies the RTC with backup power, keeping the internal clock, in the event of a main board power shut down event. **This back up battery is not a requirement in order to fully utilize all other features of the *RTC MicroSD Shield*. The shield can be used without the need of a battery if backup RTC power is not a requirement.**

Although the RTC has internal capabilities to charge certain coin batteries, the *RTC MicroSD Shield* was designed to not utilize this feature and **battery charging is permanently disabled**.

COIN BATTERIES ARE NOT A TOY. DO NOT LET YOUNG CHILDREN PLAY WITH, OR OTHERWISE HANDLE, COIN BATTERIES. UTMOST CAUTION MUST BE OBSERVED AT ALL TIMES. STORE COIN BATTERIES IN A SECURE AND ISOLATED LOCATION AWAY FROM CHILDREN WHENEVER NOT IN IMMEDIATE USE.

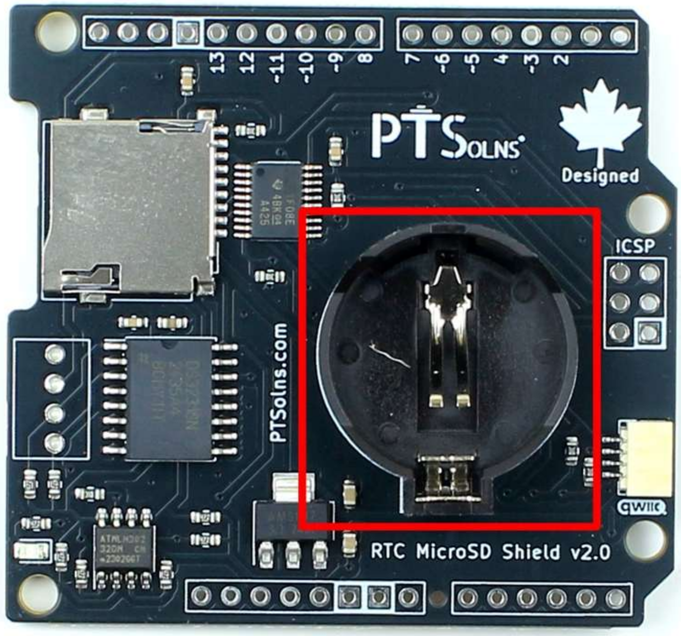

Figure 1: Coin battery holder onboard the RTC MicroSD Shield.

3.2 Micro SD Card Reader

Onboard the *RTC MicroSD Shield* is a Micro SD card reader, as shown in Figure 2. The Micro SD card should be configured as follows:

<u>On Windows</u>

Insert the Micro SD card.
Open File Explorer → Right-click the SD card → Format.
Choose FAT32 (or FAT16 if the card is ≤2GB).
Click Start.

<u>On Mac/Linux</u>

Insert the Micro SD card.
sudo mkfs.vfat -F 32 /dev/sdX

The Micro SD card reader outputs a card detect signal, which is connected to Pin D7 via the default hardware configurations. This pin can manually be changed to Pin D6 by cutting and soldering the appropriate solder jumper on the back of the board. These hardware configuration options are discussed in Section 3.6.

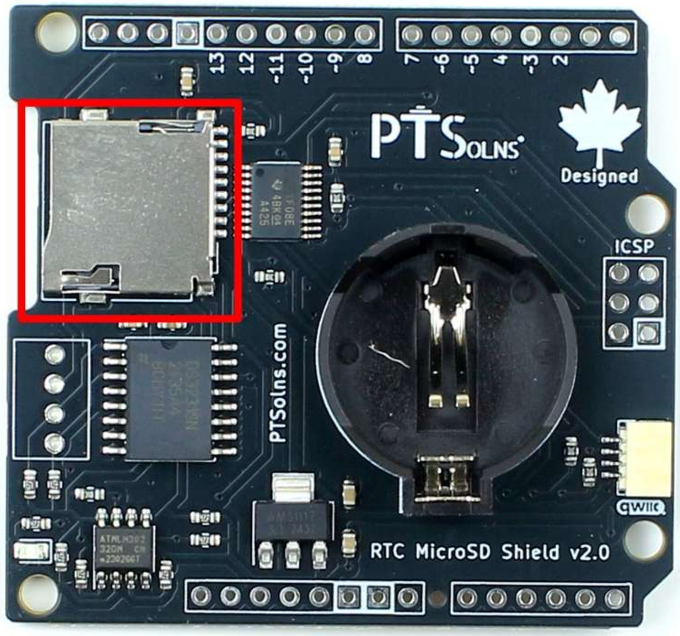

Figure 2: MicroSD card reader onboard the RTC MicroSD Shield.

3.3 EEPROM

The *RTC MicroSD Shield* offers extra non-volatile ROM via the onboard 32kBit EEPROM. The EEPROM communicates via the I2C bus and has configurable addresses as follows:

0x50
…
0x57 (default -> A0 = A1 = A2 = open)

The I2C device address can be changed by soldering the onboard jumpers A0, A1, and A2. These hardware configuration options are discussed in Section 3.6.

The EEPROM is protected with a write-protect (WP) pin, which is activated (pulled HIGH) by default. This prevents accidentally writing over critical memory in the EEPROM. In the WP mode, the EEPROM can be read, but it cannot be written to. In order to write to the EEPROM, the WP pin on Pin D8 must be manually pulled LOW in the software.

3.4 I2C, Logic Level Shifter (LLS) & Qwiic®

Onboard the *RTC MicroSD Shield* is a 5V to 3.3V logic level shifter (LLS). This predominantly is used for the Micro SD card reader, which operates on 3.3V SPI logic signals. However, the LLS is also utilized to shift the 5V I2C bus to 3.3V. This allows the board to also include a Qwiic connector, as shown in Figure 3.

Beside the Qwiic connector are pull up resistors, and a hardware configuration option to change or add to the pull up resistance. These hardware configuration options are discussed in Section 3.6.

The LLS, supplying the Micro SD card reader as well as the Qwiic connector can be enabled (pulled HIGH) and disabled (pulled LOW) via software, which is connected to Pin D5 via the default hardware configurations. This pin can manually be changed to Pin D4 by cutting and soldering the appropriate solder jumper on the back of the board. These hardware configuration options are discussed in Section 3.6. Disabling the LLS puts the inputs/outputs, including the SPI signals used for the Micro SD card reader into a high impedance state.

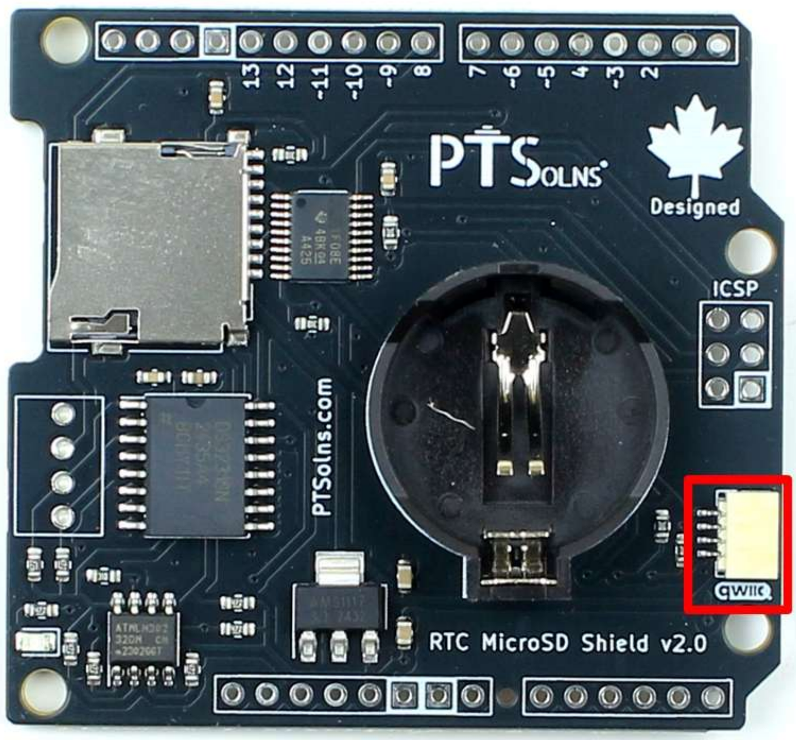

Figure 3: Qwiic connector onboard the RTC MicroSD Shield.

3.5 Compatibility & Stackability

The *RTC MicroSD Shield* is compatible with a range of 5V logic microcontroller development boards which share the common "Uno" formfactor. These include, but are not limited to, the following:

- PTSolns Uno R3+
- PTSolns DIY Uno Kit
- Arduino Uno R2 Wifi
- Arduino Uno R3
- Arduino Uno R4 Wifi
- Arduino Uno R4 Minima
- Arduino Leonardo
- Arduino Mega 2560
- Most generic 5V Uno R3/R4 clones

Furthermore, the *RTC MicroSD Shield* is fully stackable, as shown in the application example in Figure 4. The Shield is compatible with many other common Uno shields, including all members of the *PTSolns Shield-Series*.

Figure 4: Application example using RTC MicroSD Shield.

3.6 Board Configurability

The hardware configuration options on the *RTC MicroSD Shield* are available on the back of the board, as seen in Figure 5. All of the hardware configuration options are outlined in Table 1.

Table 1: Hardware configuration options.

Feature	Name	Description	Default Setting	Option via jumper
RTC (Section 3.1)	INT/SQR	Particularly useful when using INT mode to interrupt microcontroller.	D3	D2
	RST	Reset the RTC.	Closed	Open
Micro SD card reader (Section 3.2)	CD	Can be read to determine card detected in Micro SD card reader.	D7	D6
SPI for SD card	CS	Used for SPI communication with Micro SD card.	D10	D9
	COPI, CIPO, SCK	Used for SPI communication with Micro SD card. By default, only connected via ICSP header. If board does not have ICSP header, then use jumpers to connect via digital header.	ICSP header	D11, D12, D13
EEPROM (Section 3.3)	A0, A1, A2	Change I2C device address.	0x57 (A0 = A1 = A2 = Open)	A0, A1, A2 (Open or Closed)
	WP	The EEPROM is in WP mode by default. Pull LOW via software to write to EEPROM.	D8 (WP mode activated by pulling HIGH)	N/A
LLS (Section 3.4)	OE	Enable (pull LOW) or disabled (pull HIGH). Disable puts all output (SPI and I2C) in a high impedance state.	D5	D4
I2C @ 3.3V / Qwiic (Section 3.4)	Pull-Up Rs	Can be used to modify I2C pull-up strength.	10 kΩ resistors on both SDA and SCL (jumpers closed). Additional 0805 SMT footprints in parallel to existing pull-up resistors are unpopulated.	Open jumpers to disconnect 10 kΩ resistors. Additional 0805 SMT footprints in parallel to existing pull-up resistors available.
Power LED	PWR LED	Indicator power LED.	Closed	Open

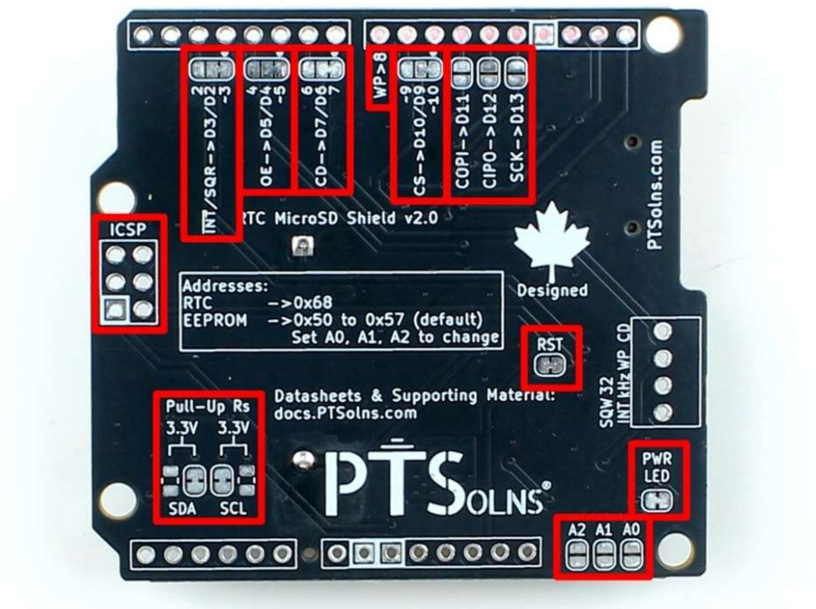

Figure 5: Hardware configuration options on the RTC MicroSD Shield.

3.7 Input & Output Signals

The *RTC MicroSD Shield* has a 4-pin port, as shown in Figure 6, which can be populated by a 2.54mm / 0.1in screw terminal. This port is an interface to input and output signals, which are as follows (from closest to the SD card reader to furthest):

CD (output)	Card detect: Can be used to read the value of card detect to determine if an SD card is inserted or not.
WP (input/output)	Write protect: Can be used to set (input) or read (output) the write protect mode of the EEPROM. See Section 3.3 for details.
32kHz (output)	32.768 kHz square wave.
SQW INT (output)	Square wave or interrupt. The RTC can be configured via software to output a square wave or an interrupt signal. If the RTC is set to SQW mode via software, four possible square wave outputs are possible: 1 Hz, 1.024 kHz, 4.096 kHz, and 8.192 kHz.

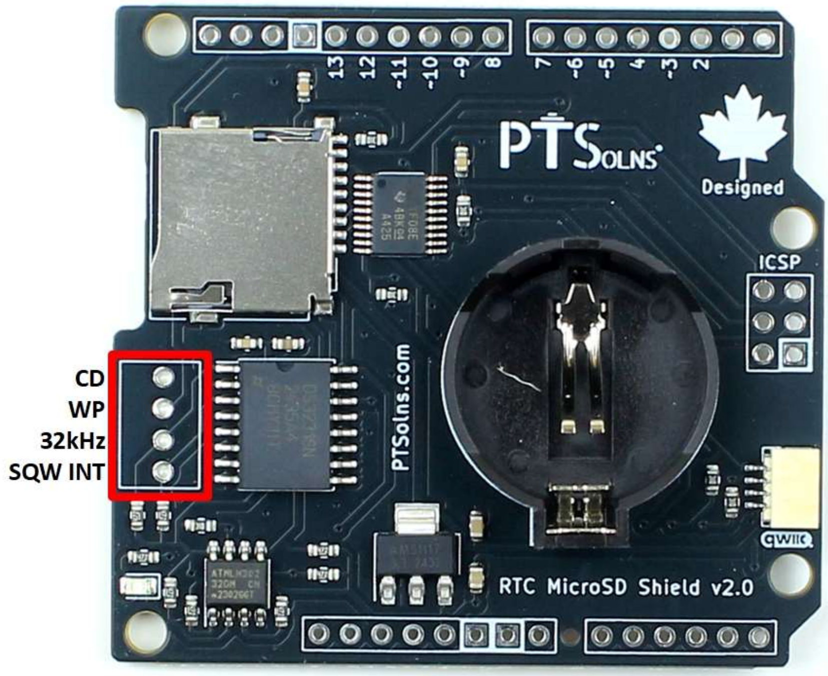

Figure 6: 4-pin signals port of the RTC MicroSD Shield.

3.8 Mark of Authenticity

Authentic PTSolns PCBs have a black solder mask color and are marked with the “PTSolns” logo in white silkscreen printing. The “Canadian Designed” symbol, consisting of the Canadian Maple Leaf with the word “Designed” underneath, can also be found on the PCB in white silkscreen printing. The “PTSolns” trademark and the “Canadian Designed” symbols are shown in Figure 7.

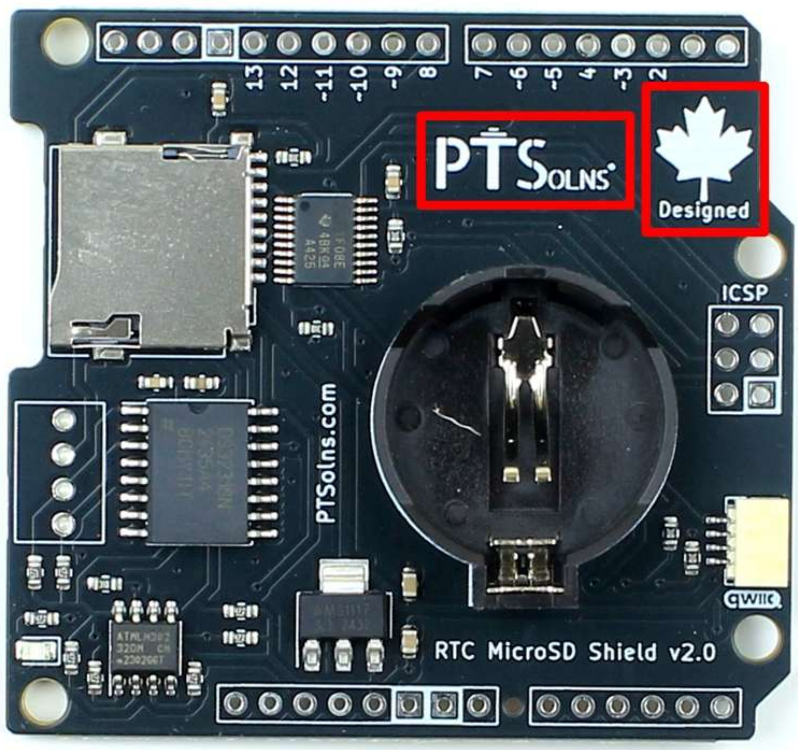

Figure 7: The “Canadian Designed” symbol and "PTSolns" trademark found on authentic PTSolns PCBs.

4 PHYSICAL PROPERTIES

The physical properties of the *RTC MicroSD Shield* are outlined in Table 2.

Table 2: Physical Properties.

	Quantity	Value	Reference
PCB	Length	57.5 mm	--
	Width	53.3 mm	**Error! Reference source not found.**
	Thickness	1.6 mm	--
	Weight (with headers)	21 g	--
	Color	Black	--
	Silkscreen	White	--
Material	Lead free HASL-RoHS surface finish		--
	FR-4 base		--
Mounting Holes	4x matching the Uno pattern		**Error! Reference source not found.**

5 USAGE AND APPLICATION

The demonstrate the various features and configuration options of the *RTC MicroSD Shield*, several example sketches are outlined in this section. All associated code can be download from the Documentation Repository

https://www.docs.PTSolns.com -> PTS-00204-211_RTC_MicroSD_Shield -> Sketches

These example sketches were written with Arduino IDE version 2.3.3.

5.1 Example Sketch: Detailed_tests.ino

This sketch is designed to give the user familiarity with the various features of the *RTC MicroSD Shield*. The following tests can be enabled in this sketch: I2C Scanner Test, RTC Test, EEPROM Test, and SD Card Test. Each of these tests have various options and detailed descriptions. By default, only the first test is enabled. The user is encouraged to go through all the tests to fully understand the various features and software and hardware configurations of the shield.

5.2 Example Sketch: RTC_Alarm_Trigger_with_Interrupt.ino

This sketch only uses the RTC onboard the Shield and demonstrates how the RTC can provide an interrupt to the microcontroller. A countdown is started and after 10 seconds, an Alarm (alarm(1)) is triggered which causes an interrupt. During the interrupt the onboard LED flashes for 3 seconds. Then the alarm is cleared, and the cycle is repeated.

5.3 Example Sketch: Sensor_Data_Logging.ino

This sketch is to demonstrate how to connect a sensor via the I2C Qwiic connector and record the data to an SD card. This example has the BMP280 sensor connected, however any similar modules can be used. It takes a temperature reading every minute and records it to the SD card.

5.4 Example Sketch: SQW_output.ino

This sketch employs the RTC to output a square wave of four hardcoded frequencies: 1 Hz, 1.024 kHz, 4.096 kHz, and 8.192 kHz. The user can change between these four outputs by restarting the microcontroller. Each time the sketch begins an integer is read from the onboard EEPROM, incremented, and saved back to the EEPROM. In this manner the restart button is used as a selection button for the four modes, cycling through the modes with each press.

The square wave output by the RTC have a peak-to-peak voltage of 5 V. The square wave signal can be connected to an oscilloscope to show the waveform, which is done in Figure 8 for the 4.096 kHz mode.

Figure 8: Example square wave output of 4.096 kHz.

5.5 Troubleshooting

This section is to help the user with common troubleshooting help.

SD card initialization failed / Not able to access the SD card

This issue is commonly due to the logic level shifter (LLS) that supplies the Micro SD card reader being disabled. By default, the LLS is disabled, and the pin OE pin has to be pulled HIGH to enable it. The default pin to enable the LLS is D5, but it can also be changed to D4 on the back of the board by cutting and closing the respective solder jumper.

See the example sketch Detailed_tests.ino and perform the test SD Card Test.

Qwiic connector not working

There could be a number of reasons why the I2C Qwiic connector is not working. The first most likely is the same as above, namely that the LLS is not enabled (see above). Another reason might be that the bus signal integrity is not good, and changing or adding pull up resistors might be required (see Section 3.4). Finally, the module itself may have issues, or the module I2C address is not working. Try this module on another board with a Qwiic connector, or try a different I2C module to ensure the issue is not with the Shield.

See the example sketch Sensor_Data_Logging.ino.

Unable to write to the EEPROM

The data on the EEPROM is protected by the Write Protect (WP) pin on D8. By default, this WP is enabled, meaning that the EEPROM cannot be written to but only read. To first write to the EEPROM ensure that D8 is pulled LOW.

See the example sketch Detailed_tests.ino and perform the EEPROM Test.

The RTC does not keep the time/date after a power cycle

If the RTC does not keep the time/date after a power cycle, likely it means the backup battery is either missing or empty. Insert a full battery CR2032 or a ML2032. See Section 3.1.1.

See the example sketch Detailed_tests.ino and perform the RTC Test.

I2C Scanner does not recognize a Qwiic connected module

The *RTC MicroSD Shield* uses a type of logic level shifter (LLS) whose primary role is to supply the SPI signals for the Micro SD card reader. The secondary role of the LLS is to supply the I2C bus on the 3.3V side. Given this, it is not ideal for I2C, in particular when using I2C scanners. The LLS introduces delays, which may interfere with rapid address polling in I2C scanners. While it works in normal I2C communication, the brief ACK pulses during scanning may not be properly detected due to slow signal transitions or insufficient drive strength.

6 REFERENCES

This section lists relevant references.

- DS3231 datasheet:
  https://www.analog.com/media/en/technical-documentation/data-sheets/ds3231.pdf

- EEPROM datasheet:
  https://ww1.microchip.com/downloads/aemDocuments/documents/MPD/ProductDocuments/DataSheets/AT24C32D-2-Kbit-I2C-Serial-EEPROM-Industrial-Grade-DS20006047.pdf

- PTSolns' Documentation Repository Sub-Domain:
  https://docs.PTSolns.com

- PTSolns website:
  https://PTSolns.com/

- PTSolns support:
  https://ptsolns.com/pages/contact

- Arduino IDE download:
  https://www.arduino.cc/en/software