RockBLOCK v2

Developer guide

Version 1.4 14th September 2016

ROCK SEVEN
LOCATION TECHNOLOGY

Table of Contents

Introduction

Thanks for choosing RockBLOCK! This developer guide refers to version 2 of RockBLOCK.

What is RockBLOCK?

RockBLOCK makes it easy to use Iridium Short-Burst Data services with your project. The PCB assembly hosts an Iridium 9602 transceiver, simplifies it's power requirements, and provides a UART interface to your project. Finally, RF considerations are taken care of by RockBLOCK's built-in patch antenna.

About Short Burst Data

SBD is a bandwidth-limited messaging system, capable of transmitting packets of up to 340 bytes, and receiving packets of 270 bytes. With a good view of the sky, it is possible to send/receive approximately once every 40 seconds.

It is suitable for applications which need to regularly send or receive small amounts of information - typically these would include tracking, telemetry, system control and monitoring applications.

It is not suitable if very low latency is required (< 1 minute), or if the data to be transmitted is larger than a few thousand bytes. Sending images, or GRIB files, is usually not sensible.

About Iridium

"The Iridium constellation of 66 cross-linked Low Earth Orbit (LEO) satellites, plus several in-orbit spares, means truly global coverage and real mobility everywhere.

The network is considered a meshed constellation of interconnected, cross-linked satellites so that each satellite "talks" with the other nearby satellites in adjacent orbits.

Unique to Iridium, this architecture provides inherent advantages in performance and reliability over other mobile satellite services providers."

(from www.iridium.com)

Importantly, Iridium operates in the microwave L-band, which means that successful data transmission relies on having an unobstructed line-of-sight view of a satellite. Satellites travel quite quickly - from horizon to horizon in under 10 minutes - so with sensible application design, it is possible to transmit data, even with a heavily obstructed view of the sky.

The Iridium website (www.iridium.com) has more information.

Rock Seven has been an Iridium Partner since 2008.

Getting Help

If you need any help at all, either with your RockBLOCK device, or with any of our associated web services, please just drop us an email.

support@rock7.com

Functional Description

Power supply

RockBLOCK includes a super capacitor as an internal energy store, which requires a 5V external power supply, which is limited to 450mA.

Alternatively, it is possible to feed RockBLOCK with a Lithium-Ion cell (3.7V), which must be capable of supplying a peak current of 1500mA.

In addition, RockBLOCK provides a regulated 5V output, suitable for powering an Arduino host, or similar.

UART

Serial communications is provided by a full UART interface at 3.3v. This can be operated in 3-wire mode (Rx / Tx / GND) with no detriment to functionality or performance. Signals are available on a 0.1" header for easy connection.

Antenna

RockBLOCK is supplied with either a 25mm square ceramic patch antenna, or an SMA jack for attaching a suitable external antenna.

USB Compatible

There is a 6-pin 0.1" header provided for direct connection to an FTDI TTL-232R-3V3 cable. This provides a virtual com port on a USB host.

Product Photographs

RockBLOCK with built-in patch antenna

RockBLOCK with built-in patch antenna

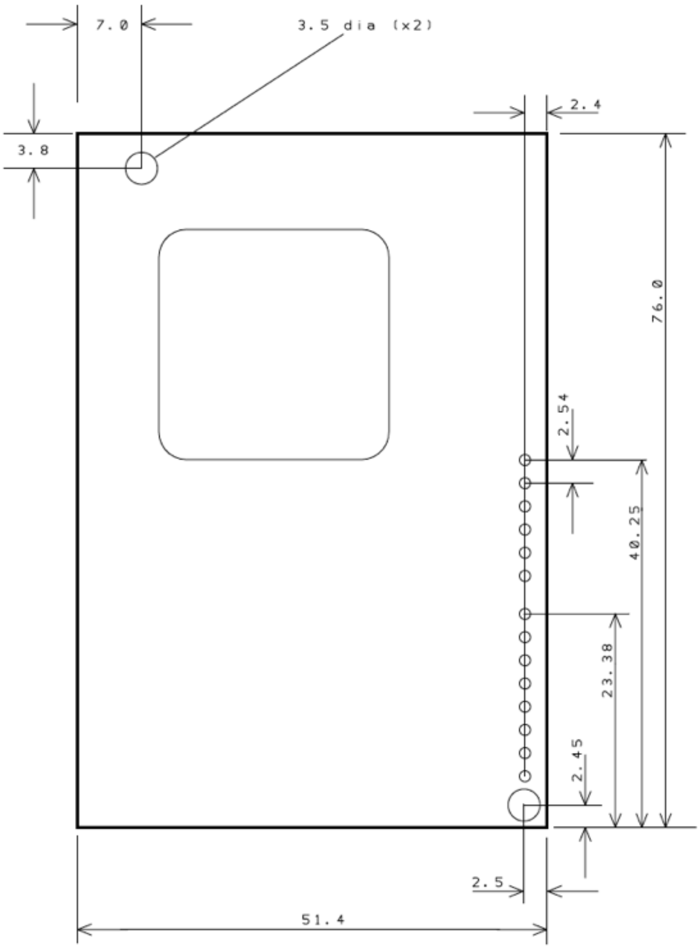
RockBLOCK with SMA connector, for use with an external antenna.

The SMA connector *replaces* the patch antenna.

Physical Specification

Dimensions

Maximum dimensions	
Length	76.0 mm
Width	51.5 mm
Height	19.0 mm



Electrical Specification

How RockBLOCK manages power

When connected to a 5V power supply, the RockBLOCK limits current consumption to 450mA.

On initial connection to a power supply, the on-board charge store must charge. At 450mA, this can take up to 25 seconds to reach full charge.

The Iridium 9602 modem will not be ‘switched on’ until the accumulated charge reaches a level at which it will be able to successfully transmit. When charging at 450mA, this is typically around 20 seconds after initial power-up.

Once charged, the idle current (with the Iridium 9602 ‘awake’) will be around 50mA. In this mode, network ‘ring-alerts’ will be received.

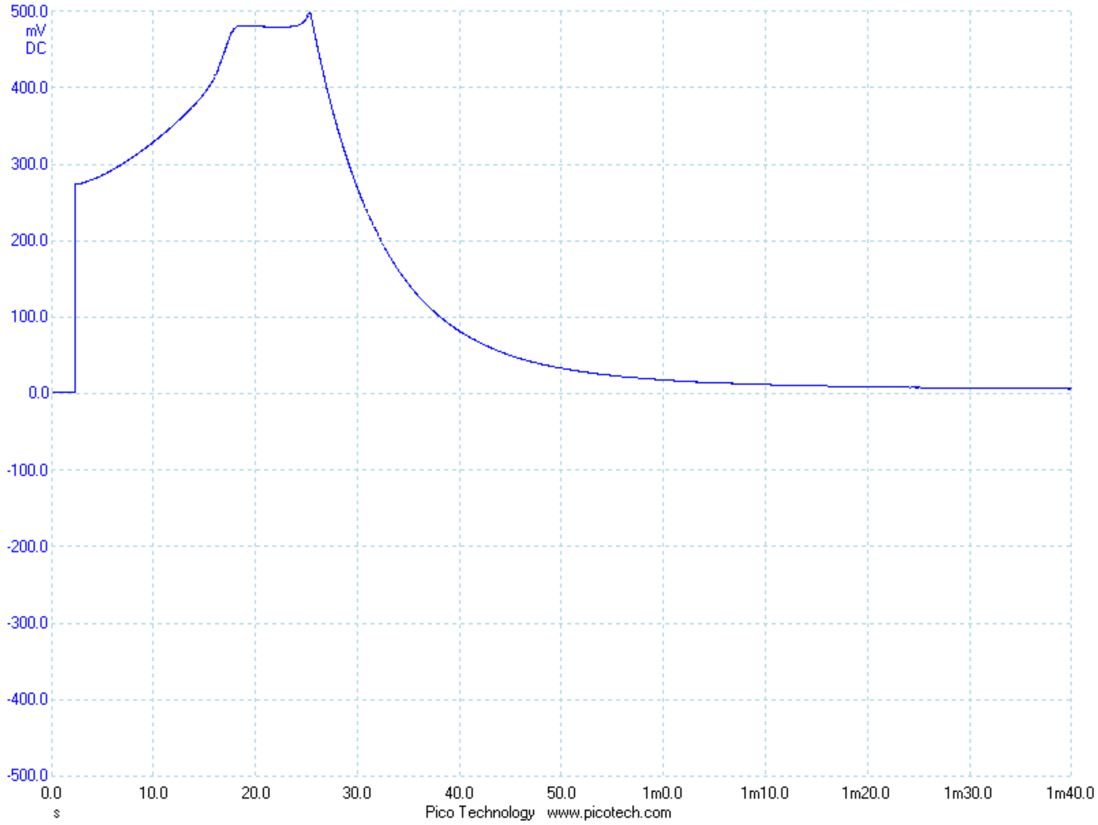
It is possible to enter a low-power ‘sleep’ mode, by switching off the Iridium 9602. This is done using the sleep pin on the direct header. In this mode, RockBLOCK will only require about 20uA. The internal charge will be maintained, so it will be possible to transmit almost immediately on wake-up.

Input Current

	Sleep	Idle	Charging (max)
5V power supply, using super capacitor	200 uA	50 mA	470 mA

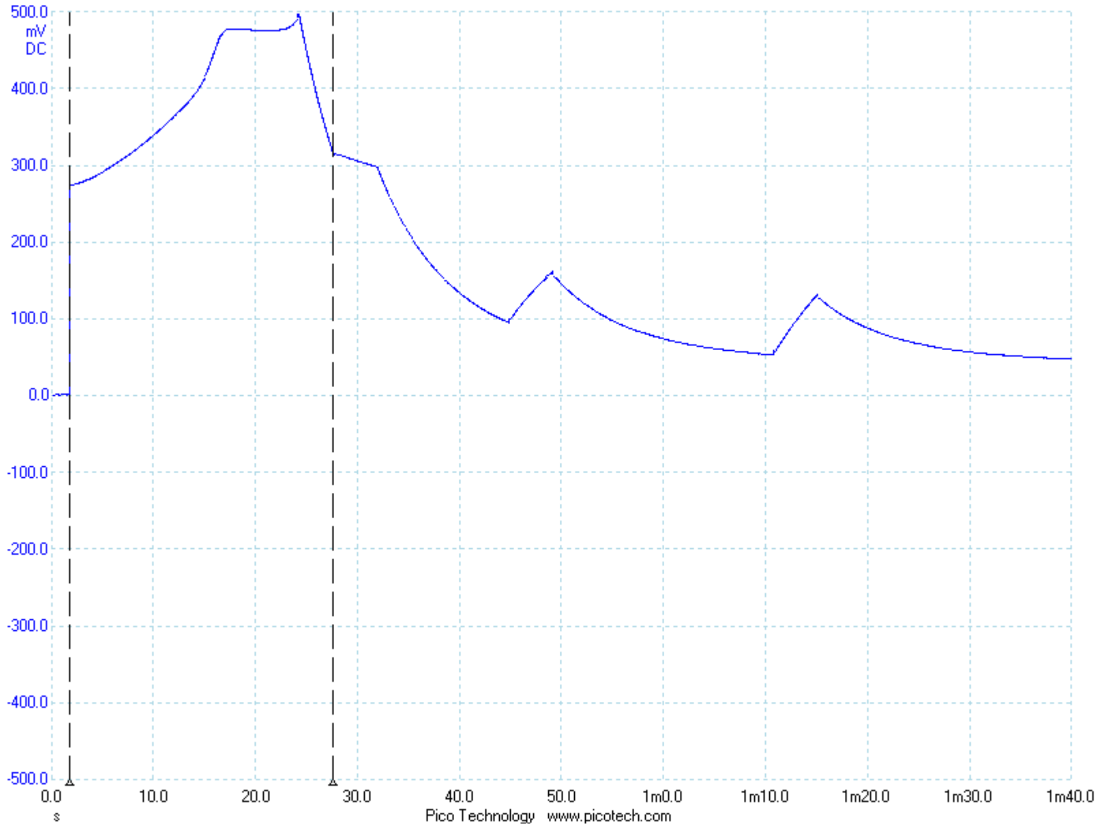
Input Current measurements

Example 1: Using a 5V power supply, and the Iridium SLEEP signal held LOW (9602 asleep), the current is measured across a 1-ohm resistor (1mA = 1mV deflection).



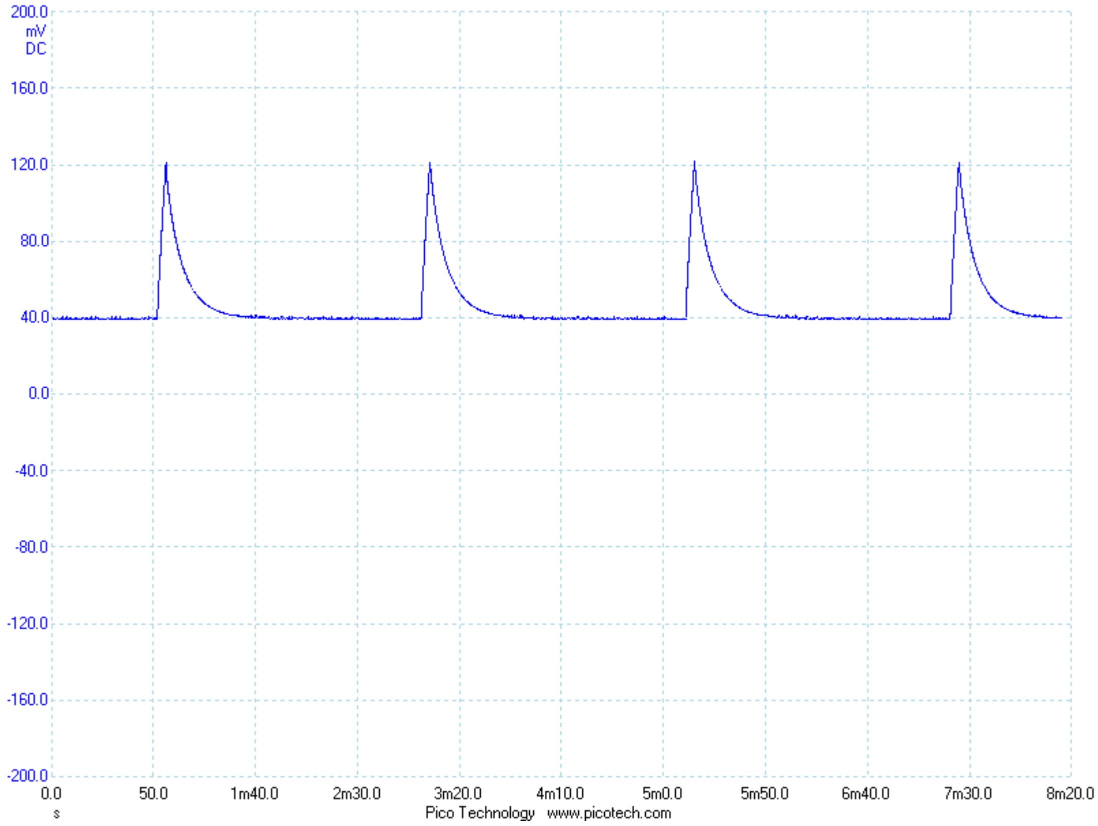
As shown in the above plot, the input current climbs to the internal current limit (approximately 470mA) over the first 15 seconds. Roughly 21 seconds after applying power, the input current starts to fall. After 50 seconds the input current is below 50mA, and it continues to fall over the next few minutes. After 15 minutes, the input current has fallen to 200uA.

Example 2: As above, but this time the Iridium SLEEP signal is held HIGH. This means that the Iridium 9602 module will switch on as soon as there is sufficient charge stored. The RockBLOCK has zero visibility of the sky. No commands are issued to the 9602, but it is evident that the module performs some regular background activity.



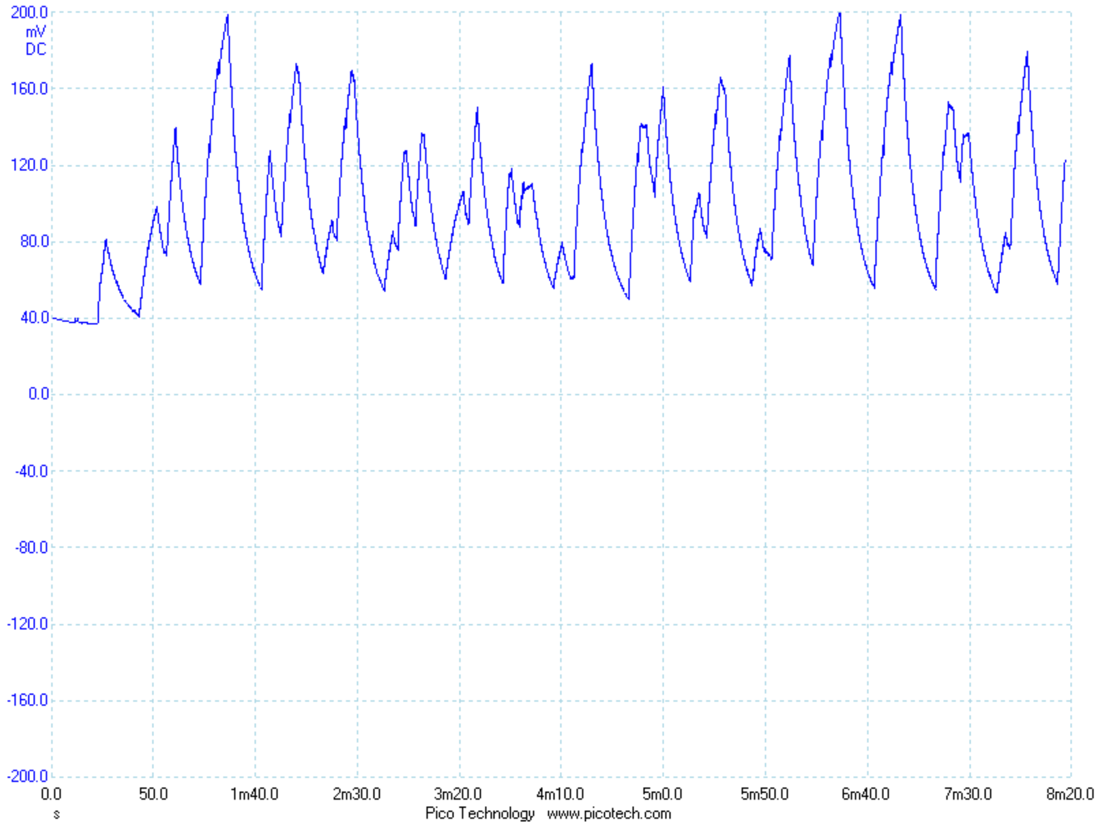
The Iridium 9602 can be seen switching on about 25 seconds after power is applied.

Example 3: Continuing from example 2, above, we leave the RockBLOCK for 15 minutes and then take another plot of input current. The RockBLOCK has zero visibility of the sky. Note that the scale on this plot is now 50s per division, and the voltage axis has also changed range.

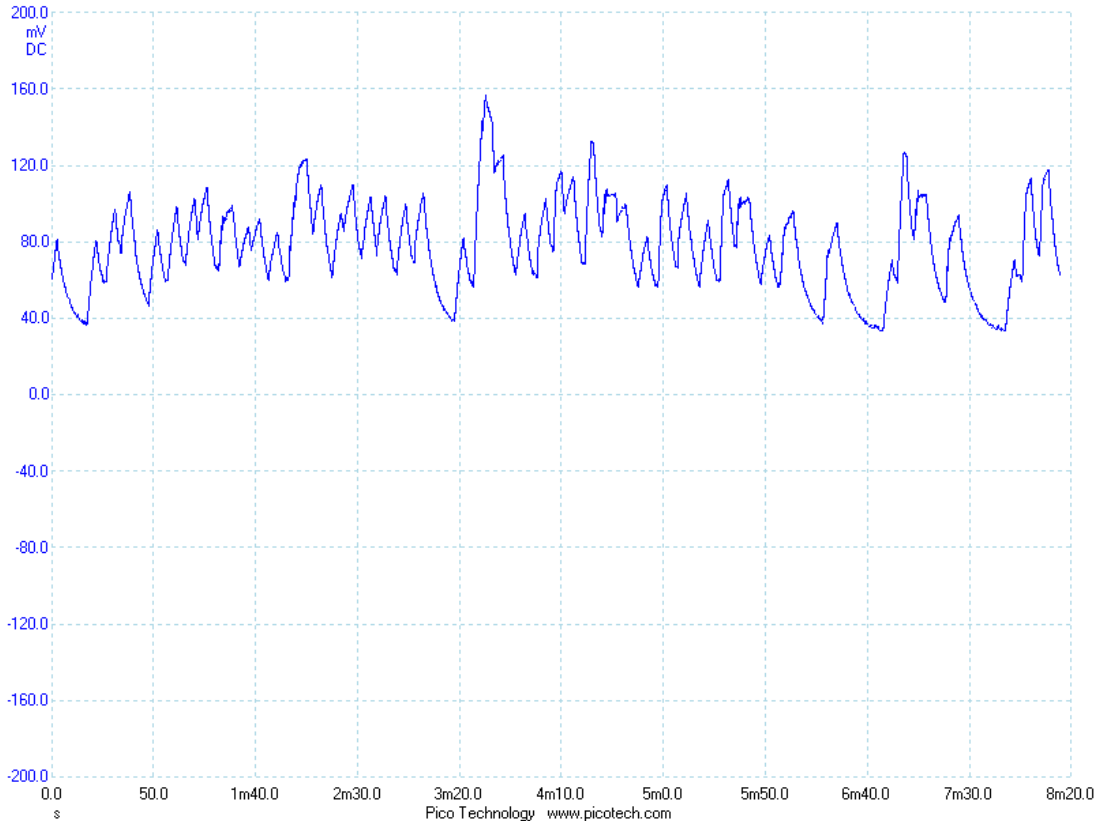


It can be seen that the Iridium 9602 is performing some regular task every 130 seconds. Average current is around 50mA. This behaviour does vary as the view of the sky changes.

Example 4: We write 50 bytes to the transmit buffer, and then initiate an SBD session every 30 seconds. With zero view of the sky (100% failure) the input current plot looks like this:



When placed in a location with about 50% visibility of the sky, the average input power requirements are lower. The following plot shows an average of 78mA over 8 minutes.



Input Voltage

	Minimum	Absolute Maximum
Using super capacitor (current limited to 450mA)	4.5 V	5.4 V
Power from Li-Ion cell	2.6* V	5.4 V

**Note that when powering from a Li-Ion cell, the source must be able to supply a burst of 7.5W for 10ms.*

Communications Interfaces

Your application will be communicating directly with the Iridium 9602 module within RockBLOCK.

Comprehensive documentation of this module is available in the *Iridium 9602 SBD Transceiver Product Developers Guide* which can be downloaded from the Rock Seven website

The data serial interface is an RS-232 9-wire interface at 3.3V digital signal levels (LVTTL) over which the 9602 and FA transfer commands, responses, and SBD message data. With respect to this interface, the 9602 behaves as a DCE (Data Communication Equipment), and the FA behaves as a DTE (Data Terminal Equipment).

If RS-232 voltage levels are needed, the FA must include an LVTTL/RS-232 level-shifter.

The baud rate can be set via the AT+IPR command. The default rate is 19200 bps.

While the 9602's data serial interface supports the control signalling of a 9-wire interface by default for those host applications which require it, the interface can be configured for 3-wire operation – where only transmit, receive and ground signals are used – with no detriment to functionality or performance.

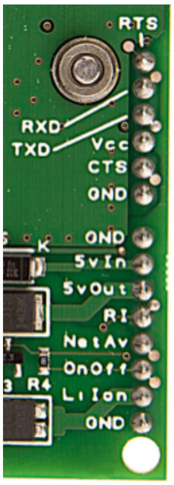
A very important note about signal directions!

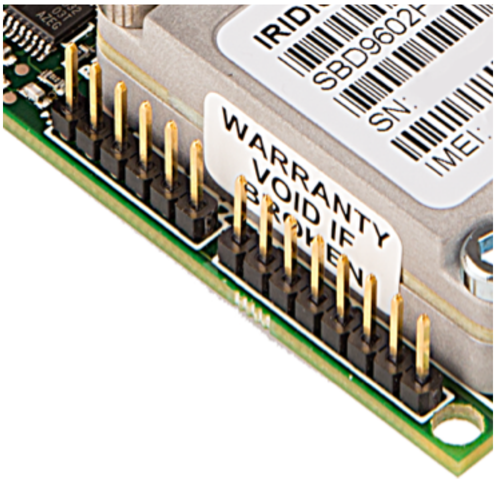
★ **TXD** and **RTS** are **INPUTS** TO the RockBLOCK

★ **RXD** and **CTS** are **OUTPUTS** FROM the RockBLOCK

0.1” Header Interface

The 0.1” header is split into 2 sections. Pins 1-6 are designed to be compatible with the FTDI TTL-232R-3V3 cable for USB hosts.





Pin	Label	Description
1	RTS	Iridium 9602 RTS
2	RXD	Iridium 9602 RX (*output from RockBLOCK*)
3	TXD	Iridium 9602 TX (*input to RockBLOCK*)
4	Vcc	5V Power supply (450mA limit)
5	CTS	Iridium 9602 CTS
6	GND	Ground
7	GND	Ground
8	5v In	5V Power supply (450mA limit)
9	5v Out	5V regulated output, for powering external Arduino host
10	RI	Ring Indicator
11	NetAv	Network available signal
12	OnOff	Sleep control
13	LiIon	3.7V Li-Ion power supply
14	GND	Ground

RockBLOCK ‘Hello World’ Example

With your RockBLOCK connected to a suitable power supply, check that your serial communications are established (default 19200, 8N1), by issuing the command:

```
AT\r
```

Which will produce the response:

```
OK\r
```

Now, these commands will transmit your first message:

```
AT&K0\r

OK\r
AT+SBDWT=Hello World\r

OK\r

AT+SBDIX\r

+SBDIX: 0, 0, 0, 0, 0, 0\r
```

The SBDIX response tell you that it was transmitted successfully, and that there were no MT messages downloaded during this SBD session.

Please refer to the *Iridium AT Command Reference* for further detail.

Note that RockBLOCK expects commands to be terminated with a **carriage return (\r)** character. This is hex **0x0D.**

Using a **line-feed (\n) 0x0A** character will not work!

RockBLOCK Gateway Commands

In general, the RockBLOCK gateway will simply transfer your MO payload directly to your chosen destination addresses (see Web Services User Guide for configuration details).

There are some specially-formatted payload that will be intercepted by the gateway to request specific functions.

Flush the MT queue

This command causes an MT message to be queued in reply, with a flag set to instruct the Iridium gateway to clear any MT messages queued, but not yet downloaded. This is very useful if you suspect that there may be hundreds of queued messages for your device, but you don't wish to download them all. Note that you are still charged for the queued MT messages, regardless of whether or not you download them.

This behaviour will be triggered if your MO payload is equal to "FLUSH_MT" (Hex: 46 4c 55 53 48 5f 4d 54).

The resultant MT reply will contain the same payload, indicating that the queue has been flushed. In this example, there were 10 MT messages queued. They were Hello1, Hello2, Hello3… etc.

```
AT&K0             (turns off flow control, as running in 3-wire mode)
OK
AT+SBDIX
+SBDIX: 0, 4, 1, 2, 6, 9
OK
AT+SBDRT
+SBDRT:
Hello1
OK
AT+SBDWT=FLUSH_MT
OK
AT+SBDIX
+SBDIX: 0, 5, 1, 3, 6, 8
OK
AT+SBDRT
+SBDRT:
Hello2
OK
AT+SBDD0
0
OK
AT+SBDIX
+SBDIX: 0, 6, 1, 8, 8, 0
OK
AT+SBDRT
+SBDRT:
FLUSH_MT
OK
```

Sending a message directly to another RockBLOCK

if you have two or more RockBLOCK configured in your account, then it is possible to have an MO message from one delivered directly to another, with the routing configured dynamically at the time of transmission.

The MO message must be formatted with a prefix, which differs depending on whether you are using binary or ASCII mode (SBDWT or SBDWB).

I order to send the message 'Hello' to the RockBLOCK with serial number 1234, using binary mode, then the MO buffer should contain:

```
52 42 00 04 d2 48 65 6c 6c 6f
```

The first two bytes decode as 'RB' and are required as a prefix. The next 24 bits (`00 04 d2`) indicate the serial number of the target RockBLOCK.

To achieve the same result using ASCII mode, the format is slightly different. Your SBDWT command would look like this:

```
AT+SBDWT=RB0001234Hello
```

Note that the serial number is required to be 7 bytes in this mode.

In both cases, the 5 or 9 byte prefix will be removed before the payload 'Hello' is delivered to RockBLOCK 1234.