



TECHNOLOGY  
SOLUTIONS<sup>UK LTD</sup>  
part of **HID**

# TSL<sup>®</sup> 3419 RAIN RFID MODULE DEVELOPER KIT USER GUIDE

[www.tsl.com](http://www.tsl.com)

Design • Development • Manufacture

# Table of Contents

<b>1. Introduction</b> .....	<b>3</b>
1.1. Equipment.....	3
<b>2. Getting Started</b> .....	<b>4</b>
2.1. Assemble the Hardware.....	4
2.2. Test the Assembled Hardware.....	4
2.3. Run a 5 Second Inventory.....	5
<b>3. TSL 3419 RAIN RFID Module</b> .....	<b>8</b>
<b>4. The 3N1X RAIN Developer Board</b> .....	<b>9</b>
4.1. Powering the Board.....	9
4.2. Data connections.....	10
4.3. Optional I/O.....	10
4.4. Indicator LEDs.....	10
<b>5. The STORM Protocol</b> .....	<b>11</b>
5.1. Anatomy of a STORM Command and Response.....	11
5.2. STORM Banks.....	13
5.3. STORM Bank Execution.....	14
<b>6. STORM Protocol Explorer</b> .....	<b>16</b>
6.1. The <i>STORM Protocol Explorer</i> Interface.....	16
6.2. The STORM Bank Controls.....	17
6.3. The Inspection Panel.....	20
<b>7. Troubleshooting</b> .....	<b>24</b>
<b>8. Regulatory Information</b> .....	<b>25</b>
Information to the User – FCC.....	25
<b>About TSL®</b> .....	<b>26</b>
<b>Contact</b> .....	<b>26</b>
<b>About HID Global</b> .....	<b>26</b>

# 1. Introduction

The TSL 3419 RAIN RFID Module is the latest, high-performance, UHF RFID product from Technology Solutions Ltd. This highly configurable and flexible module comes with a range of RF modes to ensure optimal performance in even the most challenging environments.

To aid developers in exploring the capabilities of the 3419 Module, we have provided this Developer Kit that connects to a Windows 10 PC and can be set up in minutes.

## 1.1. Equipment

The TSL 3419 RAIN RFID Module Developer Kit consists of

- 1 x RAIN Developer Board
- 1 x Region-specific Antenna Set:
  - FCC:
    - TSL P/N: **ANT016015** 902 – 928 MHz
  - ETSI:
    - TSL P/N: **ANT016015** 902 – 928 MHz (ETSI Upper Band)
    - TSL P/N: **ANT026015** 865 – 868 MHz (ETSI Lower Band)
- 1 x 15V power supply
- 1 x USB-A to USB-C data cable
- 1 x USB-A to FTDI data cable
- Software downloads<sup>1</sup>
  - This User Guide
  - The *STORM Protocol* User Guide
  - The *STORM Protocol Explorer* App in “.appx” format
  - A set of STORM command bank examples

The additional equipment you will need is:

- Windows 10 PC
- UHF RFID Tags

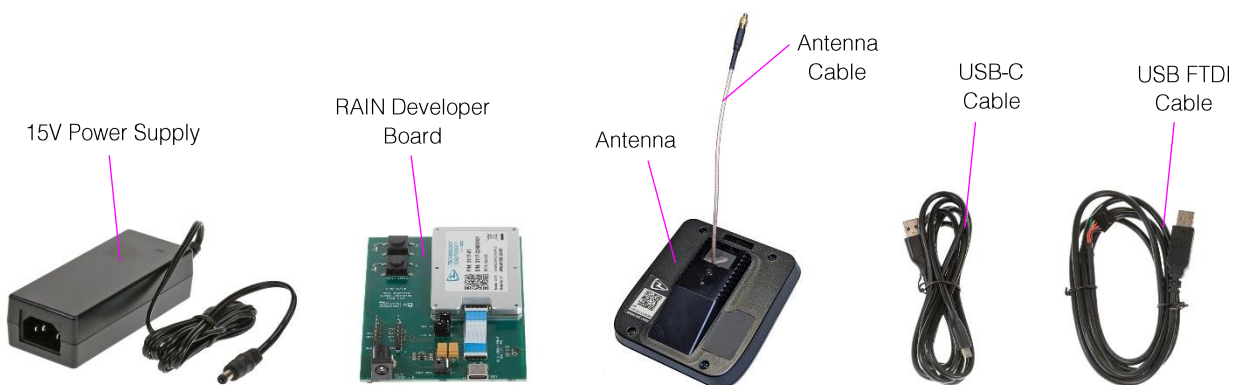


Figure 1 - The TSL 3419 RAIN RFID Module Developer Kit

<sup>1</sup> A link to the software downloads will have been provided electronically

## 2. Getting Started

This section provides a quick walk-through of setting up the Developer Kit hardware and running a few commands to test that everything is working correctly.

Firstly, install the *Storm Protocol Explorer* app to your PC by double-clicking the “*StormProtocolExplorer v n.n.appx*” package.

### 2.1. Assemble the Hardware

Assembling the Developer Kit requires just three connections (see [FIGURE 2](#)). For this walk-through we will use the USB-C interface. Set up the hardware as follows:

- Carefully push the antenna cable into the 3419-module antenna port [A] – it will ‘click’ when fully inserted.
- Power – connect the power supply to the main electricity supply and plug the barrel connector into the development board at [B].
- Insert the supplied USB-C cable into the socket marked USB [C]. Connect the other end of the USB-C cable to your PC.

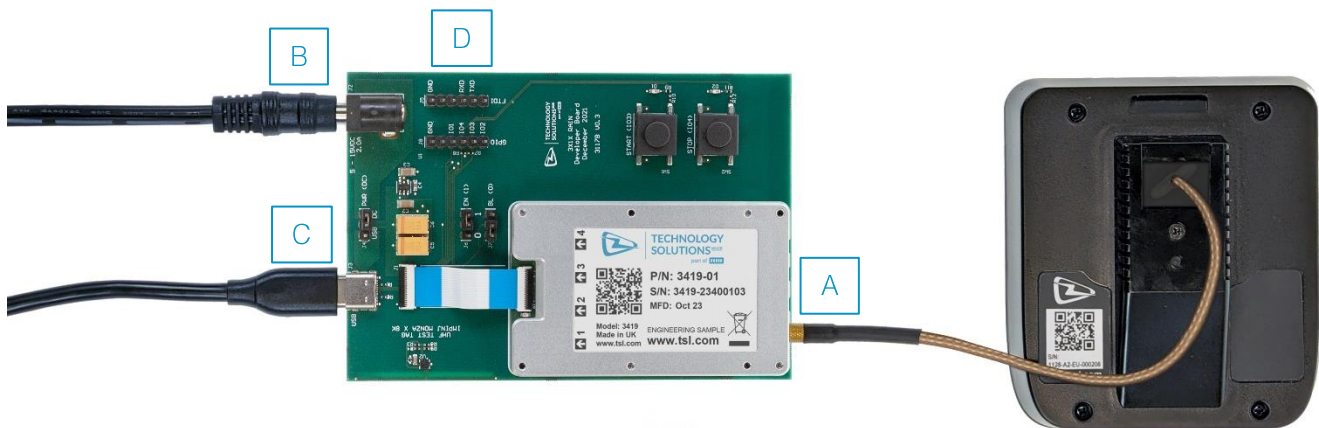


Figure 2 - The Assembled Developer Kit

Note: The module also supports a serial interface [D] that can be connected to a PC using the supplied FTDI cable.

### 2.2. Test the Assembled Hardware

Launch the *STORM Protocol Explorer* app. The UI will appear – expand the window so that all of the controls are visible, as shown in [FIGURE 3](#).

*STORM Protocol Explorer* provides a visual interface for constructing sequences of *STORM Protocol* commands, organised into *Command Banks*, and sending them to the module on the Developer Kit board. Configured *Command Banks* can be stored in a file and re-used as required. The Developer Kit comes supplied with a set of examples in the “TSL 3419 Developer Kit/Examples/” folder.

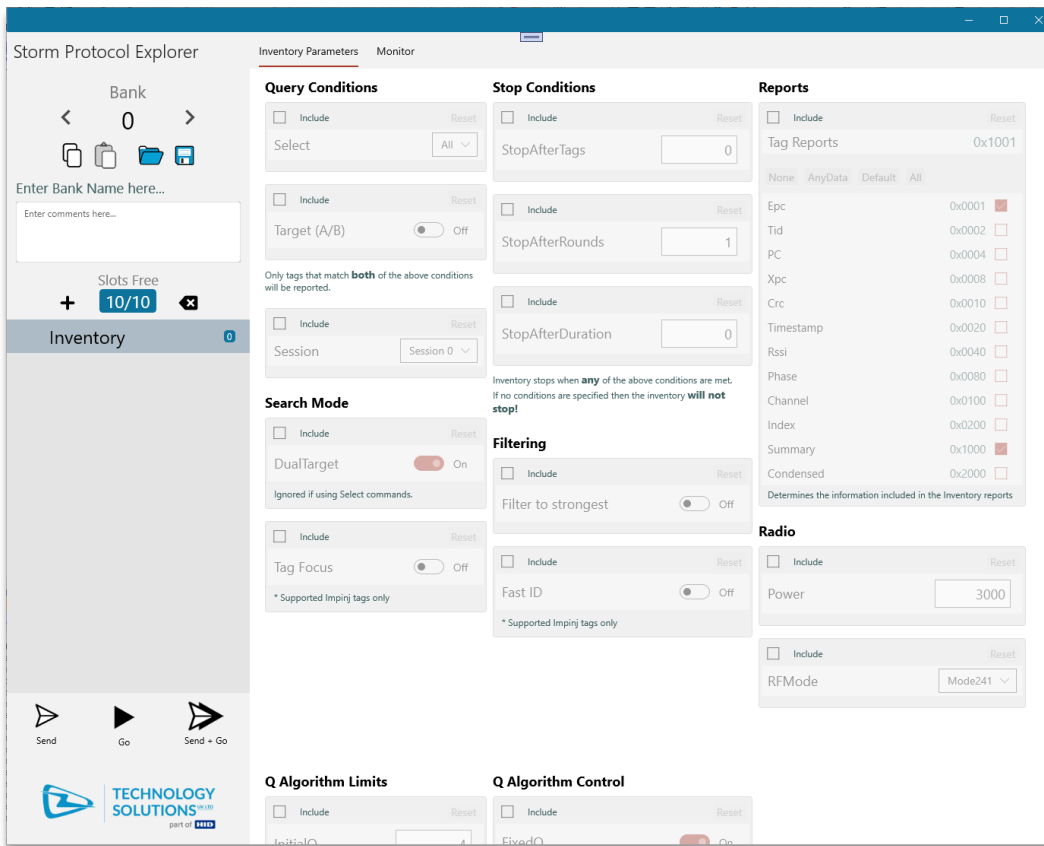


Figure 3 - STORM Protocol Explorer – First Launch

### 2.3. Run a 5 Second Inventory

With the region set, we will test the Developer Kit by loading and running the supplied 5-second Inventory.

To do this click the Load button, navigate to the “TSL 3419 Developer Kit/Examples/Getting Started” folder and load the “5 Second Inventory.jbf” file.

This Inventory parameters screen will now have various items configured but the RFMode setting may need to be modified to match the regulatory region set previously:

- Use the RFMode drop-down list to match the following:
  - a) EU RED (ETSI) region- use Mode241
  - b) FCC region- use Mode244

The Inventory Parameters should now look something like [FIGURE 4](#)

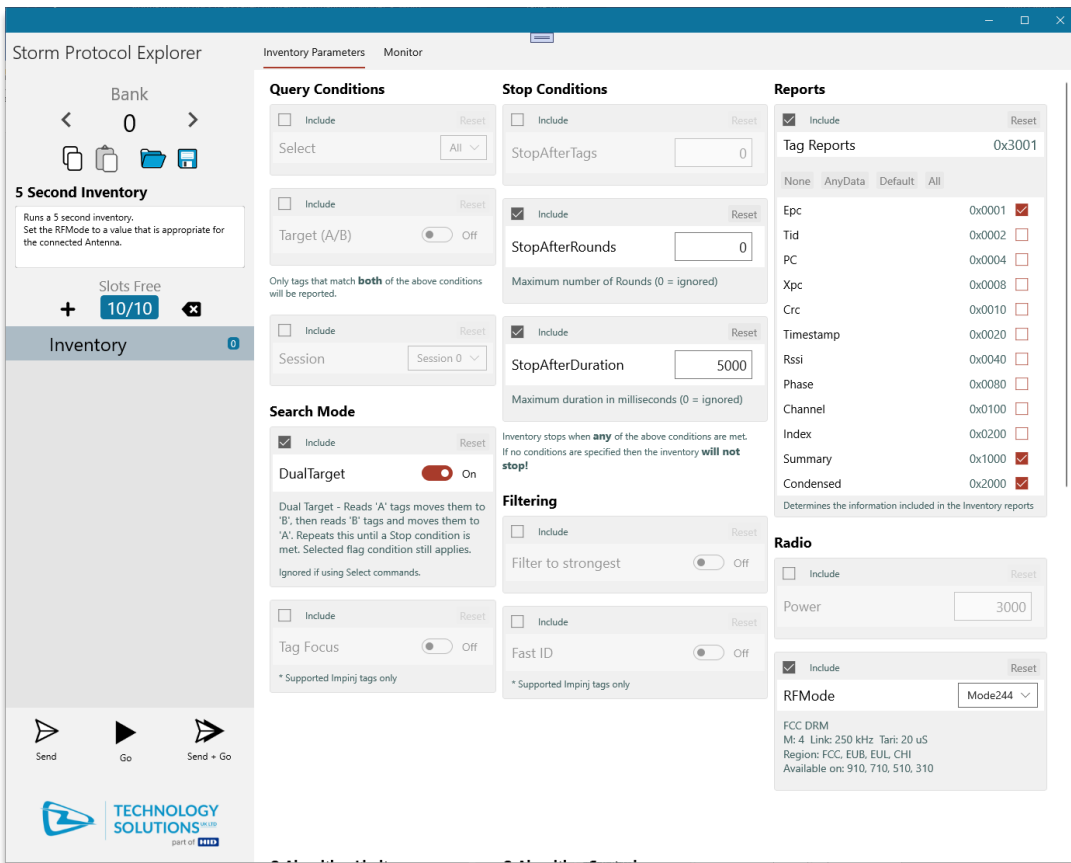


Figure 4 - The 5 Second Inventory Parameter Settings

Now, click the *Send* button – the display will switch to the *Monitor* tab where you will see, in the display area at the bottom, the commands sent to the module, in blue, and the responses from the module, in white.

You will notice that there are two commands sent - the first is a bank reset command, that ensures the bank is in a known state, and the second command configures the module’s Inventory for the current Bank (0).

With the module’s bank 0 configured, click the *Go* button to instruct the module to execute the current bank (ensure that the Antenna is facing towards your UHF RFID tags).

The display area will proceed to show the responses from all the tags that are in range of the module.

After 5 seconds, the inventory will stop and the final line should contain the end command header– “EC:” (see [FIGURE 5](#)).

A value of 0 indicates that the command has completed successfully - the full list of possible values for the EC header can be found in the STORM Protocol reference document<sup>2</sup>.

```

† TR: | EP: 8DF00000000000000007C8CCC
† SU: T=815, R=24, D=5084, S=4, P=160
† EC: 0
    
```

Figure 5 - A Successful Inventory

<sup>2</sup> “STORM Protocol for TSL 3N1X modules v 1.n.n”

Congratulations! You have just configured and executed your first *Command Bank*!

Or perhaps it didn't quite work – if so, then see the Troubleshooting section.

The following sections will introduce you to the TSL 3419 RAIN RFID Module; the Developer Board; provide a brief overview of the TSL STORM protocol and a guide to using the STORM Protocol Explorer app.

### 3. TSL 3419 RAIN RFID Module

The 3419 module is part of a family of powerful, self-contained modules that enable high performance UHF RFID reading within mobile or fixed UHF RFID reader applications.

It has a compact, lightweight design and provides industry leading RFID performance for this form factor.



Figure 6 - The TSL 3419 RAIN RFID Module

The module has many great features, some of which are highlighted below:

- Four MMCX 50  $\Omega$  Antenna ports
- The Host Interface: an 18-way, FPC connector for power and data
- Flexible Power supply requirements: 5 - 15V DC, up to 2.0A input
- Connect via USB Full-Speed CDC virtual port or High-Speed serial port (UART)
- Activity Indicator – a green LED, lit when the module is active
- Controlled using The *TSL STORM Protocol* which allows **multiple RFID operations** to be pre-configured and executed at will.
- Incorporates the new *Impinj EX10* architecture
- GS1 EPC Class 1 Gen 2 v2 compliant –with Access, BlockWrite and BlockPermalock optional command support and Impinj Fast ID and Tag Focus custom commands.
- Read rate  $\geq 1000$  tags/s
- Selectable RF regulatory regions (dependent upon antenna)
- Adjustable RF power output (1 – 30 dBm)
- A choice of DSB-ASK or PR-ASK RF modulations to adapt tag reading to multiple environments
- 10 independent *STORM Command Banks*
  - Each bank has its own set of *Inventory* round parameters and ten command-slots which may be used for *GS1 Gen2 Select* or other access commands.
- Pre-configured STORM Command Banks may be started/stopped by software or hardware triggers

Target Applications for this module are:

- Embedded handheld readers
- Fixed readers
- Low end embedded readers

Full details of the module can be found in the “3419 RAIN RFID MODULE” Datasheet



## 4. The 3N1X RAIN Developer Board

To aid developers in exploring the capabilities of the 3419 Module, we have provided this developer kit that connects to a Windows 10 PC and can be set up in minutes. At the heart of this kit is the 3N1X RAIN Developer Board – this board provides connectors for power, USB, High-Speed serial and the GPIO pins. In addition, two push buttons are provided to start and stop the execution of the configured command bank. These connections are shown in [FIGURE 7](#).

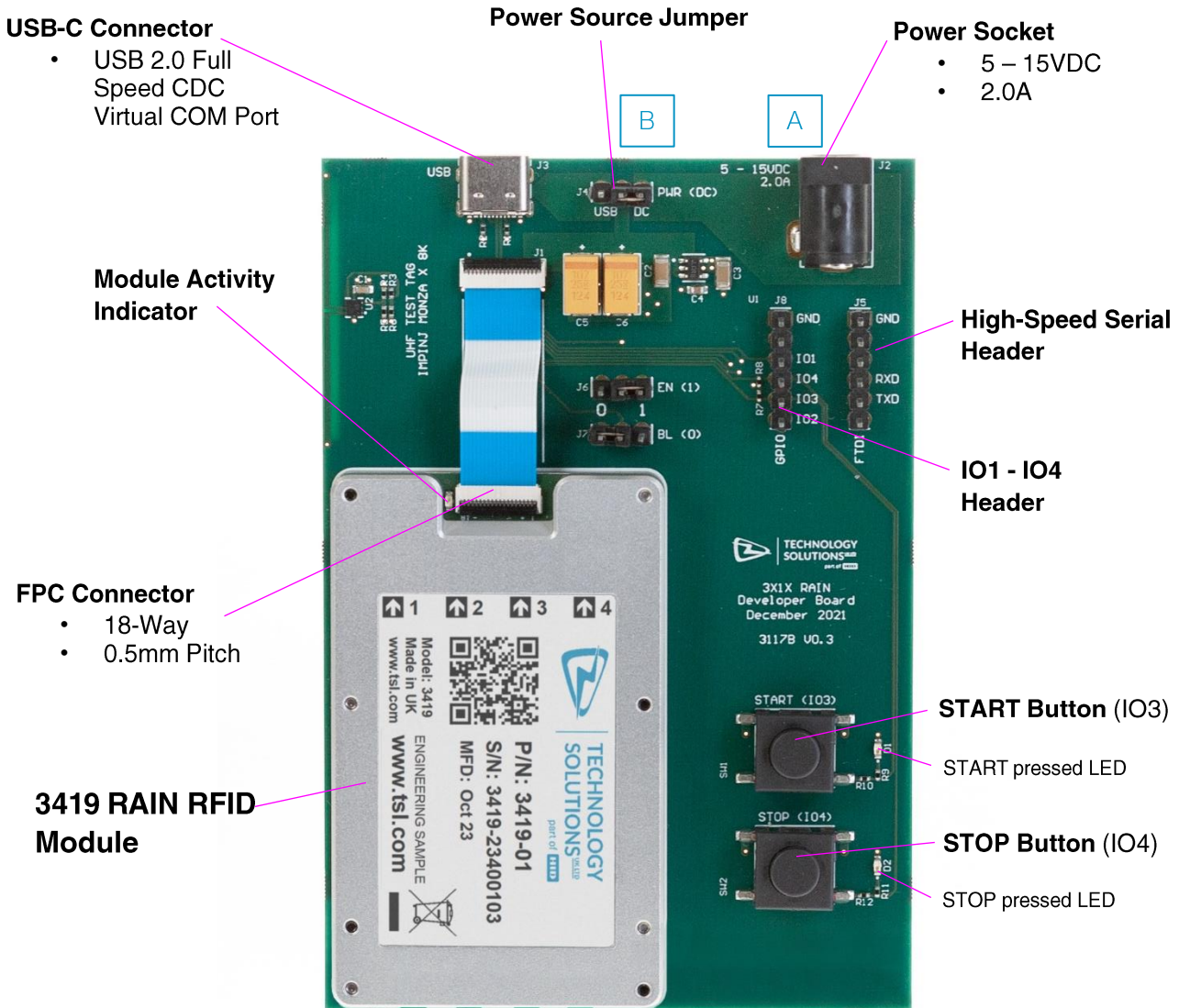


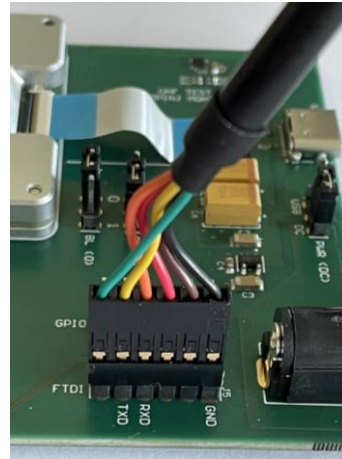
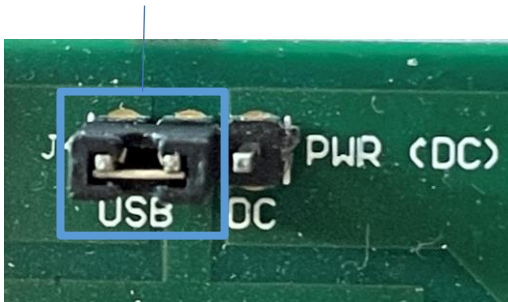
Figure 7 - The 3N1X RAIN Developer Board

### 4.1. Powering the Board

We **strongly recommend** operating the board by connecting the supplied power supply to socket [A] however, if the USB-C data connection is also capable of supplying up to 2A then the board can be operated without the power supply, by moving the power source jumper [B] as shown in [FIGURE 8](#) (Left) and the FTDI cable connection

To power the board from the USB-C cable, move the jumper to connect these two pins (marked USB)

The USB-C source must be able to supply up to 2A for correct module operation



The Developer Kit FTDI cable is not keyed – ensure the connector is inserted with the black cable on the GND pin (as shown above)

Figure 8 - Powering the board via the USB-C connector (Left) and the FTDI cable connection (Right)

## 4.2. Data connections

The 3N1X RAIN Developer Board has two data connections: 1) USB-C and 2) High-speed Serial (UART). Either of these connections can be used to control the 3419 module and they can also be used together if desired. When both serial connections are in use then the response to any commands issued will be returned on the interface that the command arrived on.

The USB-C connection provides a Full Speed CDC Virtual COM Port, when connected to a Desktop PC, and requires no additional configuration.

The UART serial connection requires the following:

- **921600** bps – default (configurable between 9600 bps - 921600 bps)
- **8-N-1**, Flow control: **None**
- 3.3V TTL Logic levels (5.0V tolerant)

The Developer Kit is supplied with a custom FTDI serial to USB cable that can be used to connect the serial port to a desktop PC – see [FIGURE 8](#).

## 4.3. Optional I/O

The developer board also provides access to the 4 x configurable I/O lines. Push buttons are wired to IO3 & IO4 as triggers for starting and stopping the inventory operation of a STORM Bank (see below). The STORM Bank controlled by these buttons defaults to Bank 0 but is configurable to any of the available STORM Banks.

IO1 & IO2 are currently reserved for future use.

## 4.4. Indicator LEDs

The developer board START and STOP buttons have corresponding LED indicators that light when the buttons are depressed.

The module is fitted with a green LED that is lit whenever the module is active.

## 5. The STORM Protocol

This section provides an overview of the *TSL STORM Protocol*<sup>3</sup>- this protocol allows a simple method of configuring the module, whilst retaining high-performance and without the complexity of running low-level RFID transponder operations.

There are, currently, two formats for issuing *STORM* commands 1) *ASCII* and 2) *Binary*. Responses will be returned in the same format (and over the same connection) as the message sent.

In this guide, we will refer to the ASCII formatted version but, bear in mind that each ASCII command has a Binary equivalent.

When using the TSL STORM Protocol SDK, commands can be constructed to communicate using the format of your choice.

In the following sections, ASCII STORM commands will be presented in a monospace font like this:

```
$ba-bnx0-re
SC: $ba-bnx0-re
EC: 0
```

### 5.1. Anatomy of a STORM Command and Response

ASCII STORM commands sent to the module have the following form:

```
$XX -T1V1 -T2V2 ... -TnVn <LF>
```

Each command must start with the '\$' character followed by a two-letter command identifier. Depending on the command, there may then follow zero or more parameters.

Each parameter is preceded by the '-' (hyphen, minus) character followed by the two-character parameter tag. The tag may then be followed by a value. Values are nearly always numeric and can be represented in the following ways:

- Hexadecimal    x7B
- Decimal        123 (or d123)
- Octal          o173
- Binary         b01111011

Spaces between parameters are optional but we'll use them in this section to aid clarity

As an example, the following command will reset command bank 0 to its default:

```
$ba -bn x0 -re
```

Here it is broken down:

**\$ba**     - command start with \$ and is the Bank command (**ba**)

**-bn x0** - bank number (**bn**) for this command is 0 (as Hex: **x0**)

**-re**     - reset the bank (**re**)

<sup>3</sup> Full details of the protocol can be found in the document: "STORM Protocol for TSL 3N1X modules, v 1.n.n"

Each command will produce a response from the module.

Each response line starts with a two-character header followed by a colon. All command responses start with the “SC:” start command header and end with the “EC:” end command header.

The start command header is followed by the command that initiated this response.

The end command header (“EC:”) is followed by an error code – a value of zero (0) indicates there were no errors executing the command.

When executing commands that return multiple tag responses, each tag response will include separate error information for the tag. So, even if every per-tag operation has failed, the command, as a whole will have succeeded and report no errors (EC: 0). Similarly, even if no tags respond to an inventory, this is still a successful command (EC: 0).

Let’s look at the response to the bank reset command from above:

SC: \$ba -bn x0 -re - The start of the response and the command that caused it

EC: 0 - the command executed without errors

Here we see the start command header with the original command followed by the end command header indicating no errors.

The response from an inventory command is a little more complex, with a report returned for each tag that responded. This report starts with “TR:” and is followed by a variable number of other headers depending on the information requested and any GS1 operations that were attempted on the tag. Refer to the STORM Protocol reference documentation for full details. When using the ASCII format this can lead to a single tag response being spread over many lines so, to prevent this, it is possible to specify a condensed tag report mode which is used in the example below:

```
SC: $ba-bnx1-go
TR: | EP: A1230000 | PC: 1000 | RI: -2872
TR: | EP: 000000000000006C7020649B | PC: 3000 | RI: -4038
TR: | EP: E2801160600020497CB711C | PC: 3000 | RI: -6783
TR: | EP: 000000000000006C8713128B | PC: 3000 | RI: -3182
TR: | EP: E2801160600020497CB710C | PC: 3000 | RI: -6717
EC: 0
```

This response includes several tag reports (TR:) and for each tag the EPC (EP:), PC Word (PC:) and RSSI (RI:) information is returned.

So, “what is a bank?” - the following section will explain...

## 5.2. STORM Banks

As described above, the 3419 Module has storage for up to 10 independent, pre-configured sets of GS1 operations and inventory parameters that are called *STORM Command Banks* or STORM Banks, for short.

Each STORM Bank consists of:

- a single set of *Inventory* parameters - configured using one or more Inventory Round (**\$ir**) commands
- Up to 10 additional operations, each allocated to an indexed *Slot*:
  - zero or more *GS1 Select* commands (**\$se**) – executed before the inventory, these are used to separate the tag population into groups of interest
  - zero or more GS1 commands – e.g. Read (**\$re**), Write (**\$wr**), Lock (**\$lo**), etc... - these are applied to each tag encountered in the inventory

The Bank (**\$ba**) command is used to:

Reset a Bank to its default state (**-re**)

Execute the Bank (**-go**)

You will notice that STORM Banks cannot be read – it is the responsibility of the connected host to track what is where!

### 5.2.1. Gen2 Slots

Each bank has 10 *Gen2 Slots* available – these can be freely mixed between *Select* operations and GS1 commands. Again, it is the hosts responsibility to track what operation is in which bank and slot.

Gen2 Slots are configured using the individual commands such as Select (**\$se**), Read (**\$re**), Write (**\$wr**) etc....

When commands are sent, they should include a Bank number (**-bn**) and a Slot number (**-gn**) – if omitted defaults of bank 0 and slot 0 will be used.

Apart from the Select command, Gen2 commands will operate on each tag encountered during the inventory operation. Select commands affect all tags in range of the module. This is described further in section 5.3.

Unlike the Inventory parameters, Gen2 Slots are configured atomically by a single command. If parameters are omitted then they will be set to the default values (as denoted in the Storm Protocol guide).

Issuing a second command to the same bank and slot will replace the current slot contents entirely.

### 5.2.2. Example of configuring a STORM Bank

Below is an example sequence of commands configuring STORM Bank 2:

```
$ba -bn 2 -re
$se -bn 2 -gn 0 -ma A1230000 -ml x20 -mo x20 -sa 4 -st 0 -sb 1 -tr 0
$ir -bn 2 -se 0 -ta 1 -dt 0 -st 1 -sr 2 -sd 0 -tr x3001
$re -bn 2 -gn 1 -db 3 -do 0 -d1 2
$re -bn 2 -gn 2 -db 2 -do 0 -d1 2
$re -bn 2 -gn 3 -db 0 -do 0 -d1 4
```

The first command issued is a bank reset command – this ensures that the bank is in a known state. This is not always necessary but when familiarising yourself with the protocol it is highly recommended.

The second line configures a Select command (**\$se**), in slot 0, to target a specific tag

The inventory parameters are next (**\$ir**) – notice that this does not require a slot number

Finally, 3 Read (**\$re**) commands go into slots 1,2 and 3

This bank is now configured to read from three different memory banks on the tag with EPC A123000. So, how do we get the module to actually do that?

### 5.3. STORM Bank Execution

After configuring a STORM Bank, such as shown in the above example, the Bank execute command (**\$ba -bn 2 -go**) is issued to start the processing of the inventory and each command in the bank.

When a STORM bank is executed, the module will:

1. Send the start command header (**SC:**) followed by the command being executed (e.g. **\$ba- bn 2 -go**)
2. Execute all Gen2 **Select** slots, *in order from lowest to highest slot number*.
3. Run the inventory operation until **any** of the Stop conditions are met, these conditions are:
  - a. the STOP trigger line is asserted
  - b. the requested number of Tags (**-st**) have been reported – or ignored if set to 0
  - c. the requested number of Inventory Query Rounds (**-sr**) have completed – or ignored if set to 0
  - d. the Inventory has been running for, at least, the requested duration (**-sd**) – or ignored if set to 0
4. For each tag that responds to the Inventory
  - a. Start a Tag Report (**TR:**)
  - b. Send the information requested in the tag reports parameter e.g. PC Word, RSSI etc..
  - c. Execute the Gen2 commands, *in order from lowest to highest slot number*.
  - d. For each Gen2 command executed:
    - i. Send a command header (e.g. **G2: 1 (RD)**)
    - ii. Report the outcome of the Gen2 command – this will vary by command but will include the Access Error Code (**EA:**) and the Backscattered Error Code (**EB:**)
5. When the inventory is complete
  - a. Issue the Summary line, if requested
  - b. issue the end command header (**EC:**) with an error number (0 if none)

The module can buffer received commands while a bank is executing but all command execution occurs in the order it was received.

So let's look at the outcome of executing a Bank command...

#### 5.3.1. Example of STORM command Bank Execution

For the STORM Bank as configured in section 5.2.2, the result of the execution would look something like this:

```
SC: $ba-bnx2-go
TR: | EP: A1230000 | G2: 1 (RD) | EA: 1 | EB: 16 | DA: 00000000 | G2: 2 (RD) | EA: 1 | EB: 16 | DA:
E2801105 | G2: 3 (RD) | EA: 1 | EB: 16 | DA: 0000000000000000
TR: | EP: A1230000 | G2: 1 (RD) | EA: 1 | EB: 16 | DA: 00000000 | G2: 2 (RD) | EA: 1 | EB: 16 | DA:
E2801105 | G2: 3 (RD) | EA: 1 | EB: 16 | DA: 0000000000000000
SU: T=2, R=2, D=63, S=3, P=39
EC: 0
```

For the inventory, this example requested tag reports in the condensed format so each component of the tag response is on the same line, separated by spaces and the “|” character.

The request for 2 rounds means that the Selected tag has been accessed twice, each time it has written a report that starts with its EPC identifier. There then follows 3 Read command responses each preceded with the “**G2: # (RD)**” header.

Each “**RD**” command response consists of the Access and Backscattered error codes and the data read from the Tag (**DA:**)

Before completing, the Summary line is issued which in this case indicates that

- 2 tags responded (T=2)
- 2 rounds occurred (R=2)
- It took 63 ms (D=63)
- The inventory stopped because of the Number of Rounds stop condition (S=3)
- The tag rate was 39 tags per second (P=39)

This has been a brief overview of some of the features of the STORM Protocol and the operation of the 3419 module. The STORM Protocol Reference guide provides all of the details for each and every module command but to provide an additional, more visual way to become familiar with the 3419 module we have produced a new application: *STORM Protocol Explorer* that we introduce in the following chapter.

## 6. STORM Protocol Explorer

*STORM Protocol Explorer* is a new app that provides a visual interface for creating and executing *STORM Command Banks*. This App will allow you to create and modify sets of commands through familiar user interface controls, optimise the parameters to match your target environment and see how the visual interface settings correspond to STORM Protocol commands.

*STORM Protocol Explorer* automatically detects the 3419 Developer board and can connect over either the USB-C or serial interfaces. Upon connection, the app queries the board for its information (**\$mi**) and displays this on the Monitor page.

The app gives a visual representation of the 3419 STORM banks and its slots. These STORM banks can be copied and pasted to provide a quick way to create multiple similar bank configurations. STORM Banks can also be loaded from and saved to file to allow the curation and sharing of bank collections for different use cases.

The *STORM Protocol Explorer* app will also remember what you were doing so that you can continue where you left off the next time you launch the app.

When a STORM Bank has been assembled and configured, a button can send the bank to the module and the bank can be executed, as many times as you like, using the Go button. There is also a convenient Send and Go button that is great for quickly testing parameter changes.

The Monitor tab provides details of the connected 3419 module and the complete over-the-wire history of the session, which can be saved for future reference. It also provides access to the STORM Protocol System commands that configure the module such as its current Region. There is even a basic entry field where ASCII STORM commands can be issued directly.

Let's begin by familiarising ourselves with the main interface components.

### 6.1. The *STORM Protocol Explorer* Interface

The *STORM Protocol Explorer* interface (FIGURE 9) is comprised of two sections 1) STORM Bank controls and 2) The tabbed inspection panel that can show [A] the selected command parameters for the current STORM Bank, [B] The Monitor pane where the connected module information controls and over-the-wire details are shown and [C] an About pane where the app version and support contact details can be found.

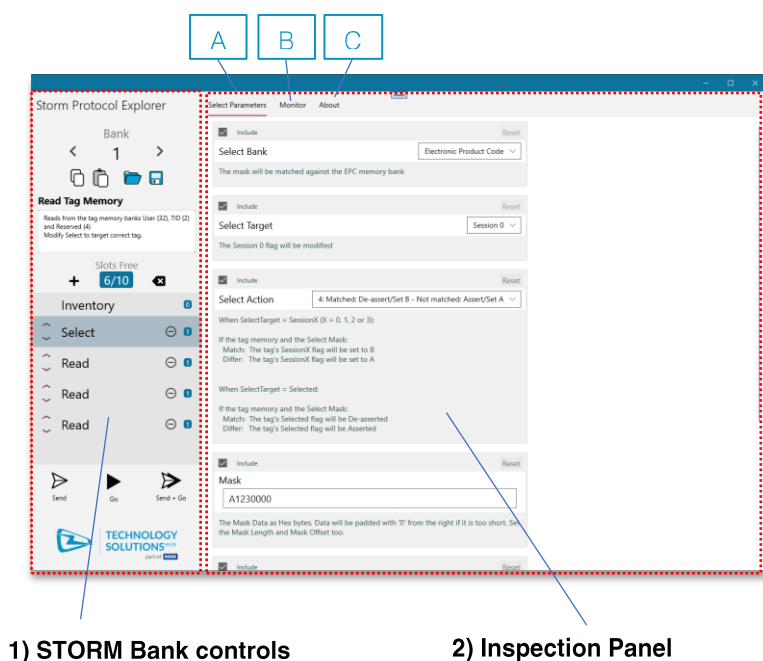


Figure 9 - The *STORM Protocol Explorer* Interface



## 6.2. The STORM Bank Controls

The STORM bank controls provide access and manipulation of the 10 banks supported by the 3419 module. The controls here allow the content of the bank to be built visually from Select and Gen2 commands and provide access to the Inventory parameters associated with each STORM bank.

### 6.2.1. Bank Selection and Manipulation

At the top of the bank controls (FIGURE 10) is the main bank selector. Use the previous bank “<” and next bank “>” buttons to switch between banks – the current bank index is shown between the controls. Below the Bank selector are the bank manipulation controls, which are from left to right:

- Copy – Copies the current bank
- Paste – Pastes the copied bank into the current bank replacing the current contents
- Open – Provides a file open dialog to load previously saved Bank files, replacing the current contents
- Save – Provides a file save dialog to allow saving of the current bank

Use these options to rapidly configure multiple banks based on existing and previous configurations.

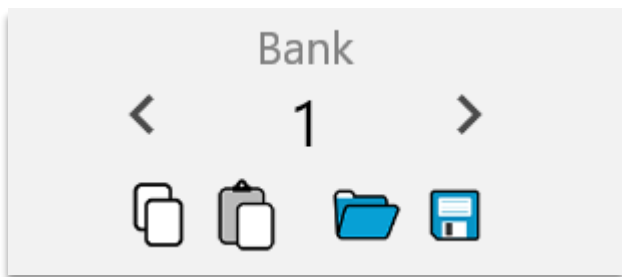


Figure 10 - Bank Selection and Manipulation Controls

### 6.2.2. Bank Name and Comments

Each STORM Bank can be given a name and comments to document the bank’s purpose. Both of these values are included when the bank is copied or saved (). These fields can be edited by clicking in them.



Figure 11 - Bank Name and Comments

### 6.2.3. The Command Slots

Underneath the Name and Comments fields are the Slot List and its controls (FIGURE 12) which provides a quick and easy method of assembling the contents of a STORM Bank. Each slot is a named Gen2 operation that correspond to a STORM protocol command. The slots are arranged in a list that will be executed in order from top to bottom when transferred to the module.

Each list has **one, non-removable Inventory command** that provides access to all the parameters needed to control the inventory and the basic tag information that is returned e.g., PC Word, RSSI etc...

The list also implements a visual organisation where the *Gen2 Select* commands will always appear above the *Inventory* slot and all other *Gen2* commands will appear below. This is a visual aid for the order of execution: Select(s) -> Inventory -> Gen2 Commands (per tag).

The position of a command in the list corresponds to the (automatically generated) slot number parameter when the bank is converted into a sequence of STORM commands.

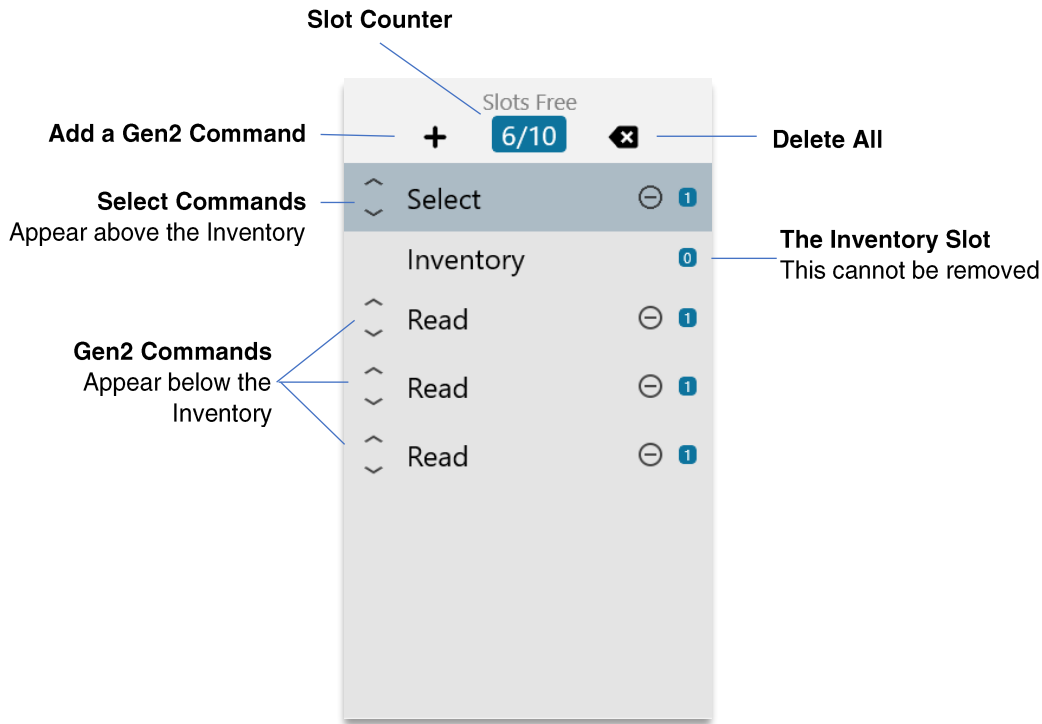


Figure 12 - The Slot List and its Controls

#### 6.2.4. Adding and Removing Command Slots

A new STORM Bank comes with just the Inventory slot but this basic operation can be expanded by clicking the Add button (FIGURE 12) to add Gen2 commands. Clicking the “+” will open a selection dialog that, temporarily, replaces the bank controls (). The *Gen2 Command Selection* dialog contains a button for each available *Gen2 Commands* – click on a button to add that command to the list.

Newly added commands will appear at the bottom of the appropriate section of the list so, *Select* commands will appear above the *Inventory* and below all other *Select* commands. The other *Gen2* commands are added at the bottom of the list.

To restore a Bank to its starting state then the “Delete All” button (✕) can be used. This will present a confirmation dialog to avoid accidental deletions.

Each STORM Bank is limited to 10 commands. The Slot List interface will ensure that this requirement is satisfied and has a “Slots Free” counter at the top of the list to track the current usage.

Each Gen2 Command slot added will adjust the free slot count appropriately – note that not all Gen2 commands correspond to a single slot on the module.

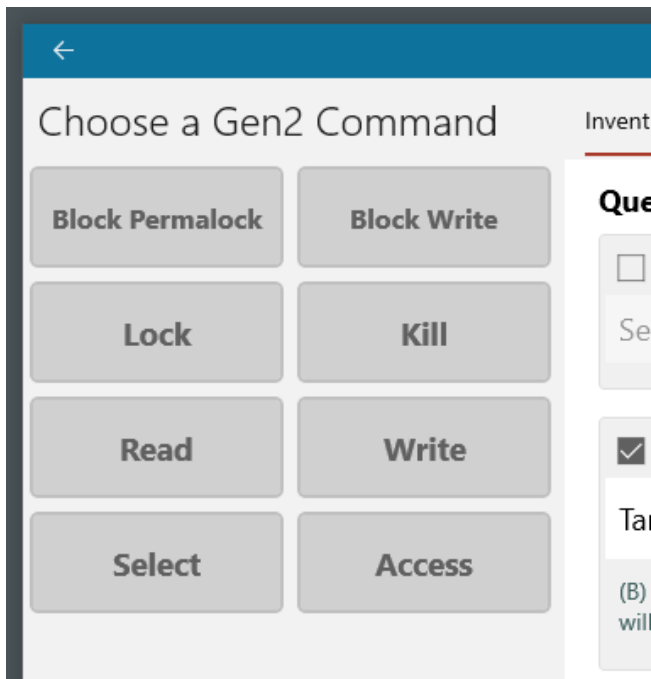


Figure 13 - The Gen2 Command Selection Dialog

### 6.2.5. Individual Command Slot Controls

Each Command slot also has individual controls (FIGURE 14) that can change execution order or remove the slot completely. An additional indicator shows the number of module slots required for the command.

Clicking the move up (^) or move down (v) buttons will move the Command Slot one slot in the chosen direction but will restrict the movement to the Command Slots section i.e., *Select* commands cannot move below the *Inventory* slot and other *Gen2* Command slots cannot move above the *Inventory* slot.

To remove an individual Command slot, click the Delete button ⊖ - a confirmation dialog prevents accidental deletions.

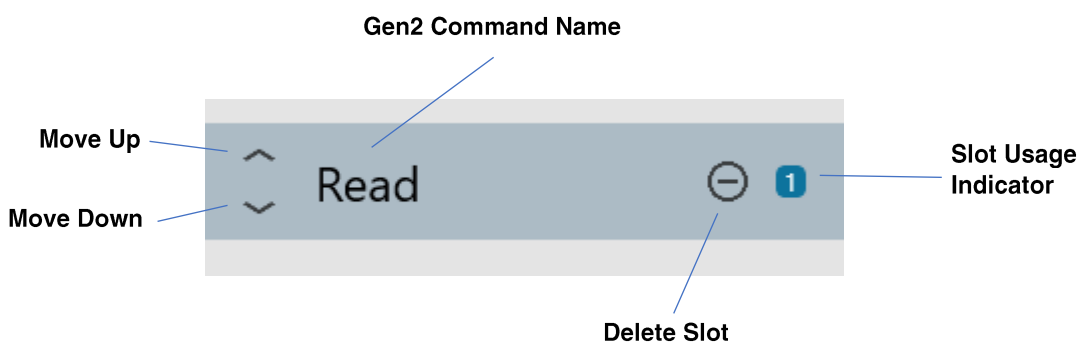


Figure 14 - Anatomy of a Command Slot

Clicking on a Command Slot will select it - this changes the content of *The Parameters View* in *The Inspection Panel* to display command-specific parameter controls, which are described below

### 6.2.6. Bank Send and Bank Go

Once a STORM Bank has been configured, there are three controls to deliver the bank to the 3419 Module and set the Storm Bank running.

The Send button (➤) will convert the contents of the current bank into a sequence of STORM Protocol commands that are then sent to the 3419 module with the outcome of these commands displayed in The Monitor View.

To ensure that all unspecified parameters have their default value, no matter how many times the Bank has been modified, the command sequence created will always start with a STORM Bank Reset command.

The Go button (▶) will send the STORM Protocol command to execute the current bank with the outcome of these commands displayed in The Monitor View.

The Send + Go button (➤▶) combines the two operations above into a single action – useful for quickly testing the effect of changes to parameter values.

When any of the above buttons are clicked, The Monitor View will be made visible to show the outcome of the operation.

## 6.3. The Inspection Panel

The Inspection Panel is home to three views 1) The Parameters View, 2) The Monitor View and 3) The About View. The sections below describe their features and purpose.

### 6.3.1. The Parameters View

The Parameters View is provided to configure each STORM Command's parameters and will also provide additional details to help understand the effect of the parameter values that have been chosen.

Whenever a Gen2 Command Slot is selected in the STORM Bank, the Parameters View changes to display a set of controls that allow modification of that command's parameters.

Each Gen2 Command parameter is presented as a tile with several common features (FIGURE 15) – which are the:

- **Name** - This is, usually, the same name as used in the STORM Protocol guide but may include spaces and capitalisation changes for additional clarity
- **Value** control – this will change dependent on the type of parameter, it may be a check box for Boolean values; a text entry field for numeric values; or sometimes, a custom control.
- **Description** – this provides an explanation of the current parameter value
- **Reset** button – this button restores the parameter to its default value
- **Include** checkbox – use this to quickly see the effect of including or omitting this parameter

Each commands parameter view may be divided into groups or include additional explanatory text.

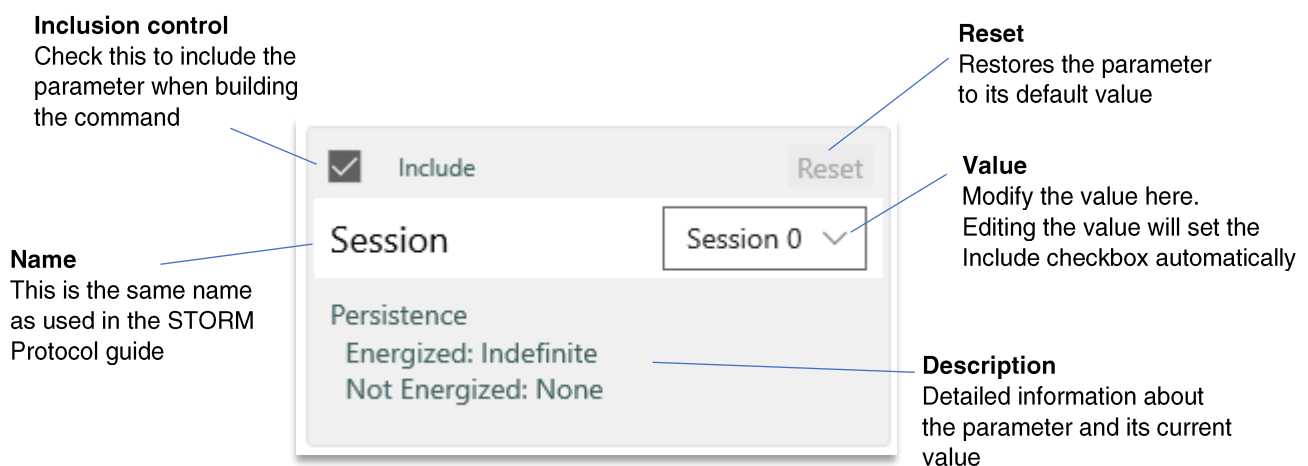


Figure 15 - A Parameter Tile

There can be many parameter tiles, their size will vary from command to command and they may be arranged into one or more groups but they all have the same basic form. In particular, there are a lot of *Inventory* parameters and the view will need to be scrolled to see them all on all but the largest monitors (FIGURE 16).

To simplify the visual complexity, parameters that are not included will be dimmed and will hide their descriptions. However, all parameters remain editable, so clicking in a *Value* field that is not currently included will automatically begin editing and, upon modification, will auto-set the *Include* checkbox.

All parameter values are retained when the *Include* checkbox is toggled so it can be just a few clicks to test commands with and without a particular parameter.

A *Reset* button is provided to return any parameter to its default value as specified in the STORM Protocol guide

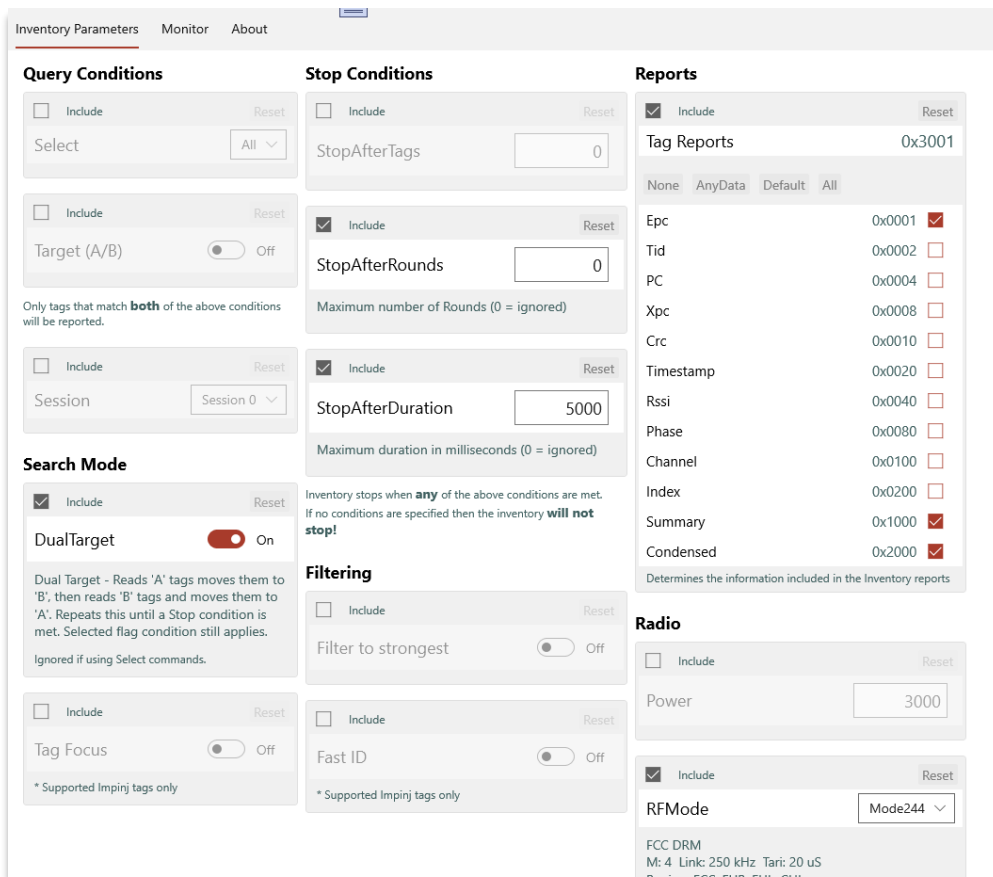


Figure 16 - The Inventory Parameters View - Scrolling (may-be) required!

### 6.3.2. The Monitor View

The *Monitor View* is divided into two areas:

- The System Area – this provides information about the connected module (if any) and provides controls to configure the Region, Serial communication lines and the IO trigger lines.
- The Communications Monitor – all over-the-wire communication is displayed here and there is a field where ASCII STORM Protocol commands can be entered manually

The Communication Monitor is where most activity occurs but the System Area also houses some important controls. The following sections describe the sections in greater detail.

### 6.3.2.1. The System Area

Whenever a 3419 module is connected, the System Area (FIGURE 17) will update with both the connection status and information about the connected module. Amongst the information provided, the module serial number, ID, it’s firmware version and its current regulatory region are among the values you may need to know.

To operate correctly the Region must be set to match the connected antenna. The *Region* pop-up menu provides the available options.

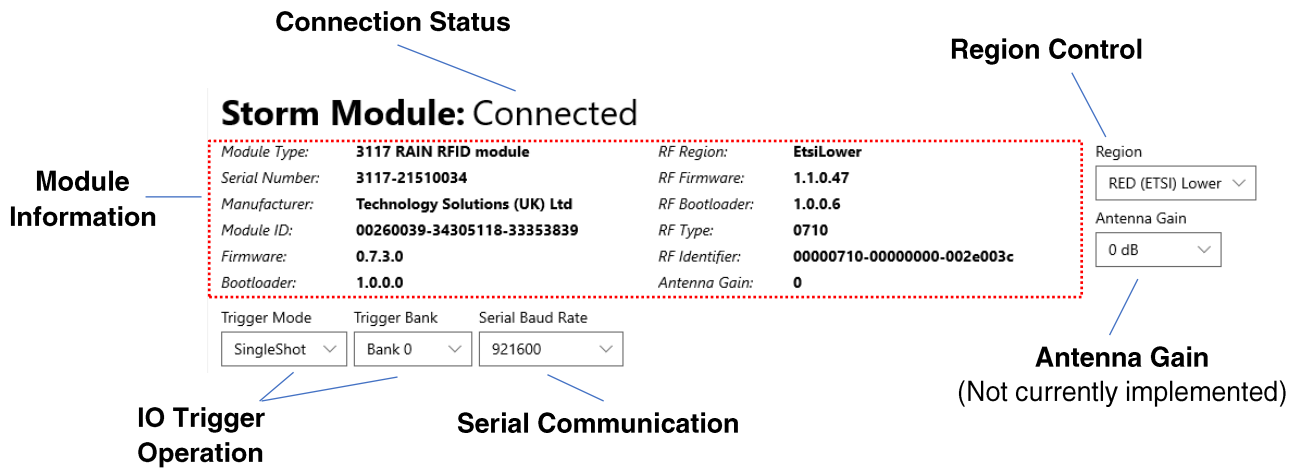


Figure 17 - The Monitor View - System Area

If you are connected to the module over the UART connection, you can set the appropriate baud rate using the *Serial Communication* pop-up menu.

The *IO Trigger Operation* controls modify the operation of the IO3 (START) line as follows:

- Trigger Mode:
  - Disabled – line has no effect
  - Single Shot – when asserted the assigned STORM Bank will be executed, the line must then be de-asserted before the bank can be triggered again
  - Repeat – While the line is asserted the assigned STORM Bank will be executed repeatedly
- Trigger Bank – set this to the STORM bank that is to be executed when the IO3 line is asserted

The 3419 Developer Board push buttons are connected to the IOS(START) and IO4 (STOP) lines with LED indicators for when the line is asserted.

### 6.3.2.2. The Communications Monitor

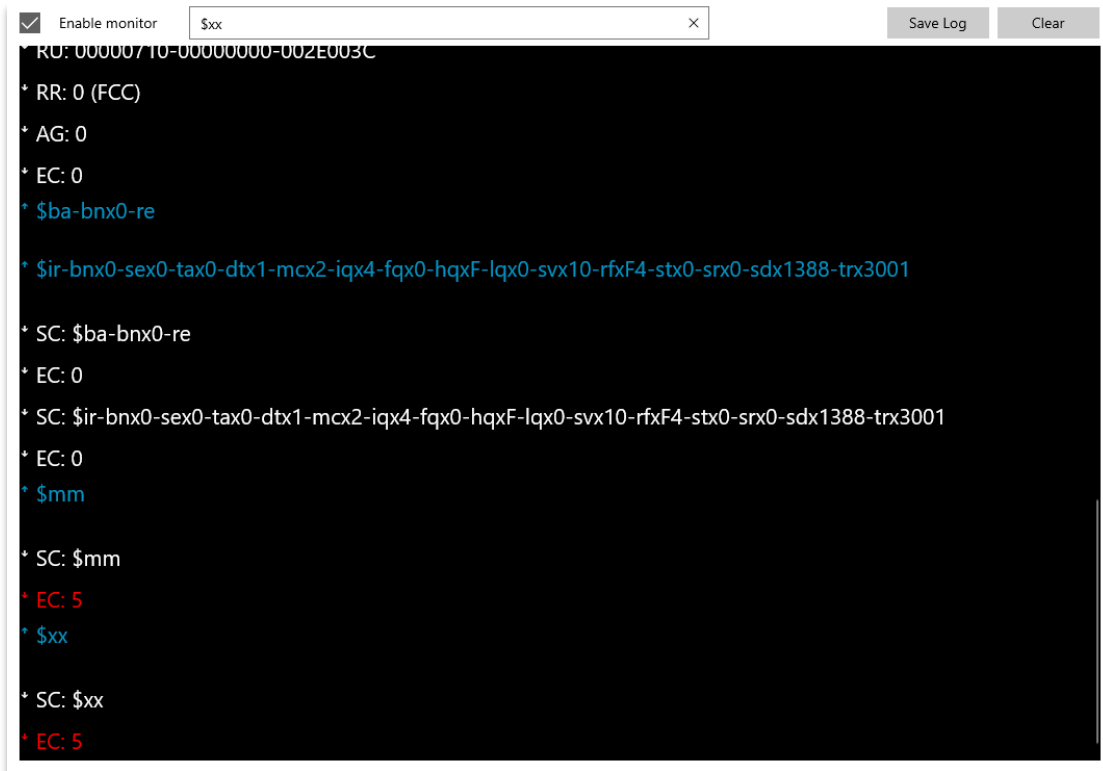
All communication to or from the module is logged and displayed in the Communications Monitor’s scrolling “Over-the-Wire” display area (FIGURE 18). The lines are formatted as follows:

- Lines sent to the module are:
  - Preceded by the send marker: ↑
  - Displayed in blue
- Lines received from the module are:
  - Preceded by the received marker: ↓
    - Displayed in white, except when containing a non-zero EC: value when they are highlighted in red

In addition to using the STORM bank controls to create and issue commands, they can also be entered manually from the Command Field. FIGURE 18 shows both valid commands to and from the module along with some manually entered “unknown” commands (EC: 5).

The *Over-the-Wire* area retains all communication for the session but this can be cleared by clicking the *Clear* button.

The entire session can be saved to file using the *Save Log* button. When saving to file, the send and receive markers (↕) are stripped out, leaving just the ASCII STORM Protocol commands.



```
✓ Enable monitor $xx x Save Log Clear
RO: 00000710-00000000-002E003C
* RR: 0 (FCC)
* AG: 0
* EC: 0
* $ba-bnx0-re
* $ir-bnx0-sex0-tax0-dtx1-mcx2-iqx4-fqx0-hqxF-lqx0-svx10-rfxF4-stx0-srx0-sdx1388-trx3001
* SC: $ba-bnx0-re
* EC: 0
* SC: $ir-bnx0-sex0-tax0-dtx1-mcx2-iqx4-fqx0-hqxF-lqx0-svx10-rfxF4-stx0-srx0-sdx1388-trx3001
* EC: 0
* $mm
* SC: $mm
* EC: 5
* $xx
* SC: $xx
* EC: 5
```

Figure 18 - Monitor View - Communications Area with a mix of Valid and Invalid commands

### 6.3.3. The About View

And finally, the *About View* contains the application version and a mailto: link if you need to contact us with feature requests, bug reports or any other feedback. It also provides a little background about TSL, the HID Global group and some legal disclaimers.

## 7. Troubleshooting

In the unlikely case that things are not working as expected try the following suggestions:

Check all the connectors are correctly inserted.

Check that the power supply plug is fully inserted and that the mains supply is on.

If you are using the serial communications port (UART) ensure the cable header is fitted correctly (see [FIGURE 8](#)).

If you received the error message shown in [FIGURE 19](#) then it is likely due to the Antenna connection. Ensure that the antenna cable is fully inserted into the module's antenna socket. Ensure that the cable is not tilted up/down or to the sides – it should be as shown in [FIGURE 20](#).

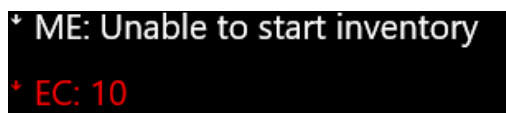


Figure 19 - An Unsuccessful Inventory

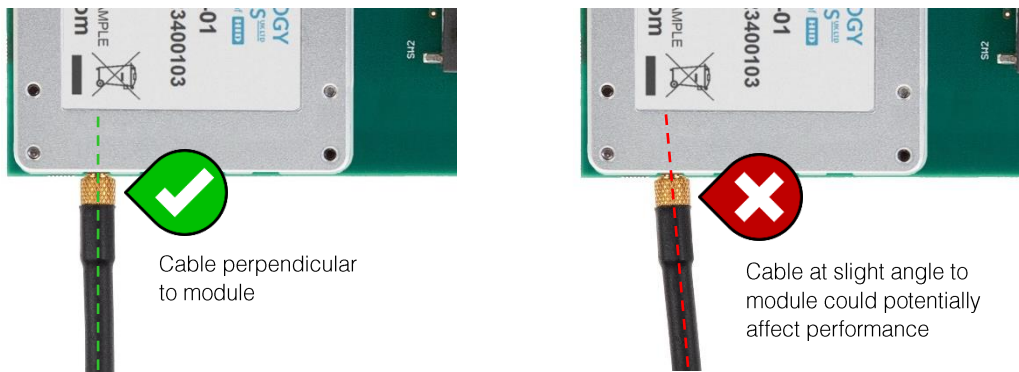


Figure 20 - The Antenna Connection



## 8. Regulatory Information

### Information to the User – FCC

NOTE: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

This device complies with FCC Rules Part 15. Operation is subject to the following two conditions:

- This device may not cause harmful interference.
- This device must accept any interference received, including interference that may cause undesired operation.

This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment. This equipment should be installed and operated with minimum distance 20cm between the radiator & your body. This transmitter must not be co-located or operating in conjunction with any other antenna or transmitter.

Any changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

## About TSL®



Technology Solutions UK Ltd (TSL®), part of HID Global, is a leading manufacturer of high performance mobile RFID readers used to identify and track products, assets, data or personnel.

For over two decades, TSL® has delivered innovative data capture solutions to Fortune 500 companies around the world using a global network of distributors and system integrators. Specialist in-house teams design all aspects of the finished products and software ecosystems, including electronics, firmware, application development tools, RF design and injection mould tooling.

TSL® is an ISO 9001:2015 certified company.



ISO 9001: 2015

## Contact

Address:	Technology Solutions (UK) Limited, Suite A, Loughborough Technology Centre, Epinal Way, Loughborough, Leicestershire, LE11 3GE, United Kingdom.
Telephone:	+44 1509 238248
Fax:	+44 1509 214144
Email:	<a href="mailto:enquiries@tsl.com">enquiries@tsl.com</a>
Website:	<a href="http://www.tsl.com">www.tsl.com</a>

## About HID Global



**HID Global powers the trusted identities of the world's people, places and things.** We make it possible for people to transact safely, work productively and travel freely. Our trusted identity solutions give **people** convenient access to physical and digital **places** and connect **things** that can be identified, verified and tracked digitally. Millions of people around the world use HID products and services to navigate their everyday lives, and billions of things are connected through HID technology. We work with governments, educational institutions, hospitals, financial institutions, industrial businesses and some of the most innovative companies on the planet. Headquartered in Austin, Texas, HID Global has over 4,000 employees worldwide and operates international offices that support more than 100 countries. HID Global is an ASSA ABLOY Group brand.

For more information, visit [www.hidglobal.com](http://www.hidglobal.com).

Technology Solutions (UK) Ltd reserves the right to change its products, specifications and services at any time without notice.