
# Octane LLRP
# version 5.2

**Proprietary and Confidential**



**http://www.impinj.com**

# Contents

# 1 Introduction

## 1.1 Purpose

This document describes the Low Level Reader Protocol (LLRP) capabilities of the Octane 5.2 software release for Speedway Revolution Readers and xArray Gateways, including Octane LLRP custom extensions.

## 1.2 Scope

This document defines Octane 5.2 LLRP. It provides a summary for system architects so they can validate and understand the standard LLRP features supported by Octane 5.2 LLRP, as well as the unique Octane LLRP custom extensions, which provide added capabilities. It provides detailed information to developers who are planning to support Impinj Readers and xArray Gateways through LLRP. Beginning with release 4.8 of this document, references to Speedway Reader have been deleted; the information in this document is now only valid for Speedway Revolution Readers and xArray Gateways.

## 1.3 References

**Table 1.1 References**

| Document | Version |
|---|---|
| EPCglobal Low Level Reader Protocol (LLRP)_ | 1.0.1 |
| EPCglobal UHF Class1 Gen2 Standard (C1G2)_ | 1.2.0 |
| Speedway Revolution Installation and Operations Guide | 5.2 |
| LLRP Toolkit Impinj Custom Extension Definition | 10.18 |

- EPCglobal Low Level Reader Protocol (LLRP):

  http://www.epcglobalinc.org/standards/llrp/llrp_1_0_1-standard-20070813.pdf

- EPCglobal UHF Class1 Gen2 Standard (C1G2):

http://www.epcglobalinc.org/standards/uhfc1g2/uhfc1g2_1_2_0-standard-20080511.pdf

## 1.4 Terms

**AccessSpec** Access Specification is a data element passed to the Reader to describe a set of operations to perform on a tag. It includes a filter set that describes the tag population to which this rule applies. In addition, it includes a list of read, write, lock, and kill commands to execute on each tag that matches the filter.

**AISpec** Antenna Inventory Specification list is contained in a ROSpec (see below), and executes in order. Each AISpec contains radio frequency (RF) parameters, inventory parameters, and duration.

**AntennaConfiguration** Each AISpec can contain one or more AntennaConfiguration parameters. These describe the RF parameters (power, frequency, receive sensitivity) and Gen2 settings (mode, filters, session) to use during an AISpec execution.

**EPCglobal** EPCglobal is an organization that leads the development of industry-driven standards for the Electronic Product Code (EPC) to support the use of RFID.

**FOV** Field-of-view is the Reader-observable world and the angular extent that is visible at a given moment. This typically relates to antenna type, number, and position.

**LISpec** Location Inventory Specification list is contained in a ROSpec. By adding a new LISpec to the ROSpec, the xArray Gateway will report the location of tags in the field of view.

**LLRP** The EPCglobal Low Level Reader Protocol (LLRP) is the industry standard.

**Location Role** An inventory in which the xArray Gateway actively detects tag location.

**LTK** The llrp-toolkit is an open source LLRP library development project.

**RO** Reader Operations is the group chartered within EPCglobal to define LLRP.

**ROSpec** Reader Operation Specification is a data element passed to the Reader to describe a bounded (start and end), triggered, and inventory operation.

**Wide Area Monitoring (WAM)** The role in which the xArray Gateway, operating as a 52-antenna reader, detects tags similar to Speedway Revolution Readers.

## 1.5 Overview

In April 2007, EPCglobal ratified the Low Level Reader Protocol (LLRP) standard, a specification for the network interface between the Reader and its controlling software or hardware. The UHF Gen 2 standard provides a standardized tag and reader radio frequency (RF) air interface protocol.

Other standards have been proposed for the controller-to-reader network interface. Why has Impinj chosen LLRP as part of its Octane software solution? LLRP is modular with respect to air-protocol. LLRP allows basic configuration and operation independent of air protocol, and supports simple configuration of readers without any knowledge of air protocol specifics. In LLRP 1.0, EPCglobal developed a parameter set to control the full functionality of Gen2 readers. For protocol-specific operations, LLRP's Gen2 parameter set provides simple access to Gen2 functionality such as read, write, lock, and kill. It also provides simple methods to select the Gen2 link parameters.

Previous standardization approaches did not go far enough to accommodate the needs of both reader and application software providers, needs that included the ability to better leverage the competitive advantages of their respective products. By creating this new LLRP standard, the advocating group led by Impinj and other RFID vendors made a rich set of vendor extension points available. These extensions provide Reader vendors with the flexibility to innovate and differentiate their products within the standardized network framework. These innovations will drive future developments of the standard.

This document is divided into sections, described below.

- Section 2 describes how to configure and establish LLRP connections with Octane.
- Section 3 describes the standard LLRP capabilities of Octane 5.2 as supported on Impinj hardware platforms.
- Section 4 specifies the Octane LLRP custom extensions available on Speedway Revolution Reader and xArray Gateway.
- Section 5 discusses advanced tag topics.
- Section 6 contains tables that define how each extension is encoded and decoded into LLRP messages and parameters.

## 1.6 Document Conventions

In this document, the term **Reader** is used to refer to both the Speedway Revolution Reader and the xArray Gateway. If the section refers to only one of these devices, the device name is used.

Throughout this document, references are made to both standard and extended LLRP messages, parameters, and fields. To help visually distinguish between these different types, Table 1.2 provides details on the conventions that are used.

**Table 1.2 Document Style Conventions**

| Type | Example | Style |
|---|---|---|
| LLRP message | IMPINJ_ENABLE_EXTENSIONS | CAPS_UNDERSCORES |
| LLRP parameter | *AntennaConfiguration* | Italics Camel Case |
| LLRP field | *ResetToFactoryDefault* | Italics Camel Case |
| Enumerated field value | 'Upon N Tags or End of AISpec' | 'Single-Quoted String' |
| File name | **'ImpinjDef.xml'** | 'Single-quoted bold' |
| LTK function | ***getLLRPStatus*** | Bold italics<br>case matches programming syntax |
| LTK class names | **CIMPINJ_TCS_RESPONSE** | Bold<br>case matches programming syntax |

# 2 Octane LLRP Connections

Octane LLRP allows both Reader- and Client-initiated connections. By default, the Reader listens for LLRP connections on the IANA-assigned TCP port 5084. Users can modify the LLRP listening port through Octane RShell or web interface. Users can enable Octane LLRP to make outgoing connections to a configurable server and port number. The address, port, and retry timers can be configured via the Octane RShell or web interface. See the *RShell Reference Manual* for more information about changing the LLRP connection configuration.

Octane LLRP accepts the first incoming connection on this port, and rejects subsequent connections as long as the first connection is active. If a connection request is received, the Reader will check the health of any existing connection. If the client TCP connection does not respond within 3.5 seconds, the Reader will automatically close the dead connection and will accept the new connection. When reconnecting after a network outage, it may take up to 3.5 seconds to accept a connection.

# 3 Octane LLRP Capabilities

Table 3.1 displays the capabilities supported by the Octane LLRP implementation, as defined by the LLRP standard. This table is organized by LLRP feature and Reader model. Not all Reader models support each Octane feature. Where relevant, the Reader reports these capabilities via the LLRP GET_READER_CAPABILTIES_RESPONSE message.

**Table 3.1 Octane LLRP Capabilities for Firmware Version 5.2.X.240**

| LLRP FEATURE | R220 | R420 | R420 + Antenna Hub | R680 | Informational Notes |
|---|---|---|---|---|---|
| **Model Name** | 2001001 | 2001002 | 2001003 | 2001007 | |
| **GPI** | 4 | 4 | 0 | 0 | These 4 GPI (referenced in LLRP as GPI 1-4). See the "GPIO Details" section in the *Speedway Revolution Installation and Operations Guide*. |
| **GPO** | 4 | 4 | 0 | 0 | These 4 GPO (referenced in LLRP as GPO 1-4). See the "GPIO Details" section in the *Speedway Revolution Installation and Operations Guide*. |
| **Antenna** | 2 | 4 | 32 | 52 | The antennas correspond to antenna ports 1-n, depending on the product capabilities. |
| **UTC (real-world) Clock** | √ | √ | √ | √ | |
| **Air Protocol Support** | 1 | 1 | 1 | 1 | UHF Class 1 Generation 2 (C1G2) |
| **Number of ROSpecs** | 1 | 1 | 1 | 1 | |

| LLRP FEATURE | R220 | R420 | R420 + Antenna Hub | R680 | Informational Notes |
|---|---|---|---|---|---|
| **ROSpec Priority Support** | 1 | 1 | 1 | 1 | Priority must always be set to 0. |
| **RFSurvey Support** | – | – | – | – | |
| **Number of AISpecs per ROSpec** | 16 | 16 | 16 | 16 | |
| **Number of InventoryParameterSpecs per AISpec** | 1 | 1 | 1 | 1 | |
| **State-Aware Singulation Support** | – | – | – | – | See section 4.2.3 for alternate control of singulation strategies. |
| **Number of Inventory Filters** | 2 | 2 | 2 | 2 | See Section 3.1.6 for details. |
| **Truncate Flag Support** | – | – | – | – | Truncate flag must always be set to 0 (unspecified). |
| **Number of AccessSpecs** | 1508 | 1508 | 1508 | 1508 | |
| **Number of OpSpecs per AccessSpec** | 8 | 8 | 8 | – | |
| **ClientRequestOpSpec Support** | | – | – | – | |
| **Number of Gen2 Modes** | Varies | Varies | Varies | Varies | Number depends on product, model, and regulatory region. Use LLRP capabilities to discover available modes. See section 3.1.4 for setting the Gen2 mode. |
| **Buffer Overflow Warning Support** | √ | √ | √ | √ | |

| LLRP FEATURE | R220 | R420 | R420 + Antenna Hub | R680 | Informational Notes |
|---|---|---|---|---|---|
| **Buffered Report Support** | √ | √ | √ | √ | See section 3.1.13 for usage details. |
| **AirProtocolInventoryCommandSettings per AntennaConfiguration** | 1 | 1 | 1 | 1 | |
| **BlockWrite Support** | √ | √ | √ | √ | |
| **BlockErase Support** | – | – | – | – | |
| **Disconnected Operation Support** | √ | √ | √ | √ | Reader will continue to execute ROSpecs and AccessSpecs when disconnected. To stop disconnected operation, disable or delete all ROSpecs and AccessSpecs before disconnecting. See section 3.1.13 for details on how events and reports are handled in this mode of operation. |
| **Set AntennaProperties Support** | – | – | – | – | |
| **TLS Encrypted Connection Support** | – | – | – | – | Only TCP connections are supported. |
| **Web Interface** | √ | √ | √ | √ | |

## 3.1 Octane LLRP Usage Notes

### 3.1.1 Octane Future Extensions

To be compatible with future versions of Octane extensions, your application must ignore all custom parameters with subtypes that it cannot understand when received at any valid LLRP or Octane extension point. In addition, it should treat any out of range enumerations as error conditions.

### 3.1.2 LLRP Response Timeout

Most commands will complete within milliseconds (nominally << 1 second). However, the following exceptions should be noted. See Section 4 for details about the timing of the Octane LLRP Custom extensions.

- GET_READER_CONFIG_RESPONSE with the AntennaProperties parameter can take up to 10 seconds, because the Reader is checking the status and connectivity of its antennas.

### 3.1.3 LLRP Message Size

Messages longer than 10 Kbytes (Speedway) or 64 Kbytes (Speedway Revolution or xArray Gateway) received by the Reader will cause a READER_EVENT_NOTIFICATION message containing a *ConnectionCloseEvent* parameter to be sent, followed by a close of the LLRP connection. The Reader transmit buffer is limited to 512 Kbytes. This corresponds to roughly 2000 TagReportData parameters per RO_ACCESS_REPORT. Client implementations should configure the ROReportSpec properly to avoid excessively large individual reports.

### 3.1.4 C1G2RFControl Parameter

Speedway Gen2 modes are selected by Impinj system engineering to provide the best performance. No Tari adjustment is necessary. Tari values passed by the client will be ignored. Octane supports automatic control and optimization of Gen2 Mode settings (Autoset) for the Reader operating environment. In previous versions of Octane software, there were separate Autoset Dense Interrogator and Single Interrogator modes. Starting with Octane 4.8, the Reader now provides Autoset modes that optimize over both environments. In addition, Octane supports several pre-configured Gen2 modes.

- A *C1G2RFControl ModeIndex* of 1000 (or 1001) configures the Reader to choose the best Gen2 link parameters for environments where the tags might be transient and we do not wish to overcommit in our search for the weakest tag.

- A *C1G2RFControl ModeIndex* of 1002 configures the Reader to choose the best Gen2 link parameters for the environments where the tags population is relatively static and we wish to attempt to search for the weakest tag.
- Link parameters reported for Autoset modes in the *C1G2UHFRFModeTableEntry* should be ignored.
- *ModeIndex* 1001 maps to *ModeIndex* 1000. For simplicity, use 1000 when referring to AutoSet.

Table 3.2 documents the official names of the Gen2 modes supported by xArray and Speedway Revolution. Modes vary depending on the model and regulatory region.

**Table 3.2 Official Octane Gen2 Mode Names**

| ModeIdentifier | Official Name |
|---|---|
| 0 | Max Throughput |
| 1 | Hybrid Mode<br>(High throughput (M=2)) |
| 2 | Dense Reader (M=4) |
| 3 | Dense Reader (M=8) |
| 4 | Max Miller<br>(High throughput (M=4))<br>Not supported by regions that support mode 5. ETSI, China, Japan, Korea, and South Africa. |
| 5 | Dense Reader (M=4) 2<br>Faster forward link than mode 2, available with regions; ETSI, China, India, Japan (916.7-920.9 MHz), Korea, and South Africa. |
| 1000 | AutoSet |
| 1002 | AutoSet static |

### 3.1.5 Gen2 Read Command

Gen2 supports a read command with zero length. This instructs the Reader to read the entire bank of Gen2 memory starting with the word indicated by the read address. The Octane releases covered by this document do not support reading with a zero word length.

### 3.1.6 Per-Antenna Configuration

LLRP supports per-antenna configuration for many standard parameters. The following parameters must be configured the same for all enabled antennas in a particular AISpec or an error will be returned.

- *C1G2RFControl* parameter
  - *ModeIndex* must be configured the same.
- *RFTransmitter* parameter
  - *HopTableID* must be configured the same.
  - *ChannelIndex* must be configured the same.
- *C1G2Filter* parameter
  - All fields and sub-parameters must be configured the same.
- *C1G2SingulationControl* parameter
  - *Session* must be configured the same.
  - *TagPopulation* must be configured the same.
  - *TagTransitTime* must be configured the same.

All other parameters can be set to unique per-antenna values.

Impinj extension parameters that control antenna settings can also be restricted in a similar manner. For details about custom parameter requirements, see the individual section that documents the extension.

### 3.1.7 LLRP Data Persistence

LLRP configuration data, including data that is set by SET_READER_CONFIG, ADD_ROSPEC, and ADD_ACCESSSPEC messages, are persistent across LLRP connections. However, a reboot of the device will reset these parameters to their default values. For information about increasing data persistence, see section 4.1.3. For information about default values for configuration data, see section 8.

### 3.1.8 LLRP Receive Sensitivity

The Speedway Revolution Reader references the RSSI sensitivity levels to an absolute sensitivity of -80 dBm. In order to set a receive sensitivity level of -47 dBm, the user must identify the ReceiveSensitivityTableEntry parameter within the GeneralDeviceCapabilities such that:

$$\text{-80 dBm} + \textit{ReceiveSensitivityValue} = \text{-47 dBm}$$

In this case, the *ReceiveSensitivityValue* is calculated to be 33, which corresponds to *Index* 25 in the Octane 5.2 LLRP capabilities. To complete the example, to set the receive sensitivity level to -47 dBm for an antenna, the *ReceiverSensitivity* field of the RFReceiver parameter should be set to 25. See Table 9.3 for the receive sensitivity capabilities of the Speedway Revolution Reader.

### 3.1.9 LLRP GPO Control

When a user disconnects from LLRP, the output state of the port pins remains as it was when connected. If the unit reboots, the GPO will be restored to the last saved GPO configuration. See section 4.1.3 for details about saving the configuration. In the absence of a saved configuration, the GPO will resort to the default configuration, as described in section 8.

### 3.1.10 LLRP AntennaEvent Parameter

The *AntennaEvent* parameter within a READER_EVENT_NOTIFICATION message reports the current connected state of the antenna. The Octane firmware can detect when antennas are connected or disconnected during inventory operation.

The Reader tracks the state of the antennas continuously and will only generate a READER_EVENT_NOTIF with the *AntennaEvent* parameter if a change is detected from the last reported status. If an antenna was previously reported to be disconnected, and a new AISpec is started, client implementations should not expect another event to be reported until the antenna is reconnected.

The recommended method for tracking antenna connectivity is to issue a GET_READER_CONFIG upon connecting to the Reader. The Reader will report the current connected state of each antenna via the *AntennaProperties* parameter. The client can then monitor the connection for any new READER_EVENT_NOTIFICATION messages that contain *AntennaEvent* parameters and update the state as appropriate. This process is done asynchronously with respect to inventory control.

With a SpeedwayR Antenna Hub connected to the Reader and the feature enabled in RShell, connected and disconnected events for the antenna are disabled by default. Enable the events with the *ReaderEventNotificationSpec* parameter in the SET_READER_CONFIG message.

### 3.1.11 LLRP Trigger Details

LLRP allows multiple types of start, stop, and report triggers. For some trigger types, additional information is required in the form of optional parameters. The LLRP specification is clear that these parameters must be present for a given trigger type. However, it does not clearly state what happens if one of these parameters appears when the trigger type does not require its presence. Octane LLRP assumes that these parameters can be present if, and only if, the trigger type requires them. As an example, if the *GPITriggerValue* parameter is present within the ROSpecStartTrigger of an ROSpec when the *ROSpecStartTriggerType* is set to 'Periodic', Octane returns an error.

### 3.1.12 LLRP Non-Specific Tag Errors

LLRP access operations (Read, Write, Kill, Lock, BlockWrite and BlockErase) all contain a result type of 'Non-Specific Tag Error' within the appropriate *C1G2OpSpecResult* parameter. Because the LLRP specification does not expose all possible C1G2 tag access error codes, the Octane firmware uses this error code as a catchall for the more specific tag errors. Table 3.3 documents the possible errors that might have occurred during tag access if the Octane firmware reports a 'Non-Specific Tag Error'.

**Table 3.3 Octane Non-Specific Tag Error Translation**

| LLRP Access Operation | Possible Specific Tag Errors |
|---|---|
| C1G2Read | CRC Error |
| | Memory Locked |
| | Memory Overrun |
| C1G2Write | Invalid Password |
| | Tag Lost |
| C1G2Kill | Tag Cannot be Killed |
| | Tag Lost |
| C1G2Lock | Memory Permalocked |
| | Memory Overrun |
| | Invalid Password |
| | Tag Lost |
| C1G2BlockWrite | Invalid Password |
| | Tag Lost |
| C1G2BlockErase | N/A (not supported) |

### 3.1.13 LLRP Buffered Events and Reports

The default configuration value for *HoldEventsAndReportsUponReconnect* is **false** for the Reader. In this mode, any events or reports generated by the Reader without a client LLRP connection are silently discarded. If a client wants to have the Reader buffer reports generated in the absence of a client connection, it must set *HoldEventsAndReportsUponReconnect* to **true**. The Reader will then internally buffer generated reports until it receives an ENABLE_EVENTS_AND_REPORTS message from the client. Upon receiving this message, all buffered reports are delivered. All future events and reports will be delivered as they are generated. Note that, in this mode, reports are buffered but events are always discarded.

### 3.1.14 LLRP TagTransitTime Field

The *TagTransitTime* field in the *C1G2SingulationControl* parameter is defined by the LLRP Specification as "…the measure of expected tag mobility in the field of view…". Internally, the Reader uses this value as part of a coarse low duty-cycle control mechanism. This means that unusually large values for this field are ignored, and the value is instead saturated at a maximum which is 10 seconds. The Reader can accept larger values, but they have no impact on the Reader operation.

It should be noted that the use of this field for low duty-cycle control is crude at best. We recommend that the low duty-cycle extension be used for precise control of Reader RF transmissions. For more information, see section 4.2.7.

### 3.1.15 LLRP ROReportSpec Parameter

The *ROReportSpec* parameter is treated as an autonomous parameter by the Reader. This varies slightly from other parameters, such as *AntennaConfiguration*, which can be decomposed. For example, if an AISpec contains an *AntennaConfiguration* parameter that does not contain an *RFReceiver* parameter, the Reader will consult the default configuration for the *RFReceiver* settings to use for that antenna. On the other hand, if an ROSpec contains a *ROReportSpec* parameter that does not have an *AirProtocolEPCMemorySelector* parameter (within *TagReportContentSelector*), or an *ImpinjTagReportContentSelector parameter*, the Reader assumes that those parameters are turned off for the subject *ROReportSpec*. The Reader does not consult the default configuration for these settings. Therefore, any *ROReportSpec* parameter that appears in a ROSpec is autonomous and complete, and will override all *ROReportSpec* settings in the default configuration.

### 3.1.16 LLRP Keepalive Messages

LLRP provides a heartbeat mechanism between the Reader, and client applications via KEEPALIVE and KEEPALIVE_ACK messages. The Reader is configured to initiate

KEEPALIVE messages via the *KeepaliveSpec* in SET_READER_CONFIG. However, the LLRP Specification does not state what action the Reader can take if its KEEPALIVE messages are not acknowledged by the client. Some versions of Octane firmware can be configured to either ignore KEEPALIVE_ACK messages, or to process these messages and use them to infer the health of a current connection. For more information, see section 4.2.12.

### 3.1.17 LLRP Transmit Power

LLRP defines transmit power as an offset into the **TransmitPowerLevelTableEntry** table for the Reader, advertised in *UHFBandCapabilities*. Because the capabilities of one product may differ from another, the absolute transmit power in dBm should not be inferred from the value configured in the *TransmitPower* field of *RFTransmitter*. For example, a *TransmitPower* index of 61 is 30 dBm on Speedway, while on Speedway Revolution the same 30 dBm absolute power is a *TransmitPower* index of 81. Client applications should always reference the advertised Reader capabilities when determining absolute power values.

### 3.1.18 C1G2 Version 1.2.0 Support

Octane 5.2 is based on LLRP version 1.0.1, which does not support C1G2 version 1.2.0. We anticipate that future versions of LLRP will add support for features that are included in C1G2 1.2.0. However, to provide access to a subset of the C1G2 1.2.0 features while the standard bodies complete their efforts on a new version of LLRP, Octane includes vendor extensions to expose the underlying air protocol features. For more information, refer to the documentation for the individual extensions.

### 3.1.19 xArray Capabilities

#### Standard Messages

##### GET_READER_CAPABILITIES_RESPONSE

If an *ImpinjRequestedData Parameter* is provided in a call to GET_READER_CAPABILITIES that specifies a *RequestedData* of Impinj_xArray_Capabilities (1003), then an *ImpinjxArrayCapabilities* parameter will be supplied in the GET_READER_CAPABILITIES_RESPONSE. For more information, see section 4.1.3.3.

#### Standard Parameters

The following standard capabilities are reported on xArray:

***GeneralDeviceCapabilities***

- Maximum number of antennas reported is 52.
- Per Antenna Receive Sensitivity Range is reported for 52 antennas.
- Air protocol supported per antenna is reported for 52 antennas.
- NumGPIs is reported as 0.
- Num GPOs is reported as 0.

***LLRPCapabilities***

CanDoTagInventoryStateAwareSingulation is reported as **False.**

***WAM mode only:***

- *MaxNumSpecsPerROSpec* is reported as 32.
- *MaxNumAccessSpecs* is reported as 1508.
- Otherwise, *MaxNumSpecsPerROSpec*=1

**Note**: Access operations are not recommended on the xArray.

***RegulatoryCapabilities***

- The maximum TransmitPowerTableEntry reported is +30.0 dBm.

**Custom Parameters**

***ImpinjRequestedData***

The existing *ImpinjRequestedData* parameter has been extended to allow for the specification of xArray capabilities and configurations.

Use GET_READER_CAPABILITIES to get the following xArray capability values.

| Value | Description |
|---|---|
| 2003 | 'Impinj_xArray_Capabilities' |

Use GET_READER_CONFIG to get the following xArray configuration values.

| Value | Description |
|---|---|
| 2010 | 'Impinj_Tilt_Configuration' |
| 2011 | 'Impinj_Beacon_Configuration' |
| 2012 | 'Impinj_Antenna_Configuration' |
| 2013 | 'Impinj_Location_Configuration' |

For more information about the *ImpinjRequestedData* parameter, see section 4.3, which describes LLRP custom parameters.

#### *ImpinjArrayVersion* Parameter

The existing *ImpinjDetailedVersion* parameter is extended in xArray to report a new *ImpinjArrayVersion* parameter as a custom extension.

##### Description

This parameter provides detailed information about the individual components that comprise the integrated antenna array. The meaning behind each field varies by Reader model.

**Table 3.4 ImpinjArrayVersion Field Descriptions**

| Field | Description |
|---|---|
| SerialNumber | The serial number of the antenna array hardware. |
| FirmwareVersion | The firmware version of the antenna array beam forming component. |
| PCBAVersion | The hardware version of the antenna array PCBA component. |

##### LLRP Dependencies

This custom parameter has no LLRP dependencies.

##### Allowable Extension Points

- *ImpinjDetailedVersion* parameter

#### Definition

Definitions for the *ImpinjDetailedVersion* parameter are shown in Table 3.5.

**Table 3.5 ImpinjArrayVersion Parameter Definitions**

| ***ImpinjArrayVersion* Parameter** |
|---|
| **SerialNumber**: UTF-8 String<br>**FirmwareVersion**: UTF-8 String<br>**PCBAVersion**: UTF-8 String<br>**Custom Extension Point List**: List of <custom parameter> [optional] |

### *ImpinjxArrayCapabilities* Parameter

This parameter is used to return capabilities information about the array functionality of an xArray Gateway. This parameter can be requested by specifying a *RequestedData* value of 1003, *Impinj_xArray_Capabilities*, in an *ImpinjRequestedData* parameter supplied to a call to the GET_READER_CAPABILITIES command.

#### LLRP Dependencies

None

#### Allowable Extension Points

- GET_READER_CAPABILITIES_RESPONSE message

#### Definition

Definitions for the *ImpinjxArrayCapabilities* parameter are shown in Table 3.6.

**Table 3.6 ImpinjxArrayCapabilities Parameter**

| ***ImpinjxArrayCapabilities*** | |
|---|---|
| **Parameter** | **Definition** |
| **SupportsLISpecs** | Boolean - true if Location role operations are supported (always 1 in Octane 5.2) |
| **SupportsTISpecs** | Boolean - true if Transition role operations are supported (always 0 in Octane 5.2) |
| **SupportsTSISpecs** | Boolean - true if Transition Singlerole operations (always 0 in Octane 5.2) |

# 4 Octane LLRP Configuration

Octane extends LLRP with custom extensions to provide critical functionality unique to Impinj Reader and xArray Gateway products. These features utilize the custom extension mechanism provided by LLRP. Table 4.1 summarizes the Octane LLRP custom extensions and a description of the features usage.

For each Octane LLRP custom extension, the documentation includes a description of the feature, a discussion of LLRP dependencies, the allowable extension points for the extension, and the definition of API elements. The subsections below outline the information provided for each Octane LLRP custom extension and its relevance to the developer or system architect who want to use Octane LLRP:

**Description**: The description subsection contains specific information about the extension, including what it does and how to use it. The description contains the high-level information required to implement the extension.

**LLRP Dependency**: The LLRP Dependency subsection describes how the contents of the extension affect other standard LLRP fields and parameters. Many extensions provide additional functionality over what standard LLRP offers. The settings in the standard version of the protocol elements can be modified or overridden entirely by the presence of an extension parameter. Where applicable, the LLRP dependency section clarifies the behavior.

**Allowable Extension Points**: The allowable extension point subsection describes where the extension is permitted within the LLRP messaging structure. Not all parameters in LLRP allow the presence of custom parameters. The LLRP specification documents the allowable locations of custom extensions.

Octane further restricts each individual custom extension and where they may appear within LLRP messages. Each custom parameter (not applicable for custom messages) lists the LLRP extension points at which the parameter may appear. Octane LLRP custom extension parameters can appear in any order in an LLRP custom extension point.

**Definition**: The definition subsection of each Octane extension defines the fields and sub-parameters that make up the extension. Field types and definitions for enumerated values are included in this section.

**Table 4.1 Octane Custom LLRP Extension Summary**

| Octane LLRP Extension | R220 | R420 | R640 | R680 | Sections | Summary Description |
|---|---|---|---|---|---|---|
| **Enable Extensions** | √ | √ | √ | √ | 4.2.1<br>4.2.2 | Required to utilize any of the Octane LLRP custom extensions features. |

| Octane LLRP Extension | R220 | R420 | R640 | R680 | Sections | Summary Description |
|---|---|---|---|---|---|---|
| **Detailed Version Information** | √ | √ | √ | √ | 4.3.8 | Provides detailed version information for the subcomponents that make up the Reader. |
| **Sub-Regulatory Region Control and Reporting** | √ | √ | √ | √ | 4.3.2 | Used when regulatory regions offer several distinct modes of operation or when a single Reader offers multiple regulatory regions. The response to this command can take up to 10 seconds, because the Reader must reconfigure itself for the new region. |
| **Inventory Search Mode** | √ | √ | √ | √ | 4.3.3 | Configures the inventory algorithms for optimum performance. This is an alternate method to the *StateAwareSingulation* parameter in LLRP that requires detailed Gen2 knowledge. |
| **Fixed Frequency Lists** | √ | √ | √ | √ | **4.3.4** 4.3.5 | Allows the client to control and configure automatic frequency selection for regulatory regions with fixed frequency operation. |
| **Reduced Power** | – | √ | √ | √ | 4.3.6 | Provides Reduced Power operation level configurability on certain channels within the FCC regulatory region. |

| Octane LLRP Extension | R220 | R420 | R640 | R680 | Sections | Summary Description |
|---|---|---|---|---|---|---|
| **Low Duty Cycle** | √ | √ | √ | √ | 4.3.7 | Provides clients the ability to configure a low duty cycle mode to limit interference. The Reader manages the duty cycle based on tag observation statistics. |
| **Save Settings** | √ | √ | √ | √ | 4.2.3 4.2.4 | Allows the application to save configuration settings in the Reader. The response to this command can take up to 2 seconds, while the Reader commits the configuration to persistent storage. |
| **GPI Debounce** | √ | √ | √ | – | 4.3.9 | Configures the minimum period between general-purpose input (GPI) transitions reported by the Reader. Debounce allows the Reader to be directly connected to mechanical switches or other "noisy" inputs. |
| **Advanced GPO** | √ | √ | √ | – | 4.3.10 | Allows for more advanced use of the Reader general-purpose outputs (GPOs). GPOs can be pulsed for a specified duration, or can be tied to a specific Reader operational status. |
| **Temperature Reporting** | √ | √ | √ | √ | 4.3.11 | Allows for polled reporting of the internal Reader temperature. |

| Octane LLRP Extension | R220 | R420 | R640 | R680 | Sections | Summary Description |
|---|---|---|---|---|---|---|
| **Link State Monitoring** | √ | √ | √ | √ | 4.3.12 | Configures the Reader to monitor the state of a LLRP connection using the LLRP KEEPALIVE mechanism. |
| **Report Buffer Behavior** | √ | √ | √ | √ | 4.3.13 | Instructs the Reader on how to buffer reports it sends to client applications. It can be used to decrease latency of tag reports at the expense of both Reader and Client CPU utilization. |
| **Access Spec Configuration** | √ | √ | √ | Don't Use | 4.3.14<br>4.3.15<br>4.3.16 | Allows for fine-tuned control over AccessSpec execution including the number of words sent over the air interface during a BlockWrite operation, and how many times an operation is retried before declaring failure. |
| **C1G2 BlockPermalock** | √ | √ | √ | √ | 4.3.17<br>4.3.18<br>4.3.19<br>4.3.20 | Exposes the C1G2 air protocol BlockPermalock operation. |
| **QT Technology** \|trade\| | √ | √ | √ | √ | 4.3.21<br>4.3.22<br>4.3.23<br>4.3.24 | Allows the Reader to access and configure the QT Technology \|trade\| of the Impinj Monza 4QT tags. For more information about this feature, reference the Monza 4QT datasheet. |

| Octane LLRP Extension | R220 | R420 | R640 | R680 | Sections | Summary Description |
|---|---|---|---|---|---|---|
| **Serialized TID** | √ | √ | √ | √ | 4.3.25<br>4.3.26<br>4.3.32 | Allows the Reader to report both the EPC and TID as part of normal inventory, without the need for an explicit AccessSpec. |
| **RF Phase Angle** | √ | √ | √ | √ | 4.3.25<br>4.3.27<br>4.3.33 | Reports the RF phase angle of the communication with the tag over the air interface. |
| **High Resolution RSSI** | √ | √ | √ | √ | 4.3.25<br>4.3.28<br>4.3.34 | Reports the peak power of the tag backscatter in a higher resolution than is available via LLRP. |
| **GPS Location** | √ | √ | √ | √ | 4.3.25<br>4.3.29<br>4.3.35<br>4.3.38<br>4.3.39<br>4.3.40 | Allows the Reader to report its GPS location when attached to a supported GPS-capable device. The GPS location can be obtained instantaneously, or included within tag reports. |
| **Optimized Read** | √ | √ | √ | √ | 4.3.30 | Allows for the reporting of additional tag memory content during inventory without the use of AccessSpecs. The reads are optimized by the Reader for enhanced performance. |
| **AISpec Looping** | √ | √ | √ | √ | 4.3.37 | Allows the Reader to execute AISpecs repeatedly. |

| Octane LLRP Extension | R220 | R420 | R640 | R680 | Sections | Summary Description |
|---|---|---|---|---|---|---|
| **Intelligent Antenna Mgmt** | √ | √ | √ | √ | 4.3.41 | Allows for the Reader's Intelligent Antenna Management Feature to be disabled. Enabled by default this feature ensures that the reader uses an antenna only if tags are detected on it |

## 4.1 Standard Messages

The following subsections describe standard messages that are special cases supported in Octane LLRP.

### 4.1.1 xArray Messages

**GET_READER_CONFIG**

The xArray Gateway supports AntennaID values from 0 to 52 in commands where this parameter is specified. As with standard LLRP usage, values 1 to 52 address the 52 individual antennas (beams), while 0 is used to address all antennas.

The following values for *RequestedData* are **not supported**, and any GET_READER_CONFIG that specifies these values are ignored:

- 9 - GPIPortCurrentState
- 10 - GPOWriteData

The *RequestedData* field of the *ImpinjRequestedData* parameter has been modified to allow requests of configuration information specific to xArray. For more information, see section 4.3.1.

**GET_READER_CONFIG_RESPONSE**

- GPOWriteData parameter will not be returned.
- GPIPortCurrentState parameter will not be returned.

## 4.2 Custom Messages

The following subsections describe the custom messages supported in Octane LLRP.

### 4.2.1 IMPINJ_ENABLE_EXTENSIONS Message

This top-level extension custom message is used to enable the exchange of all other Impinj extensions. By default, all of the Impinj extensions are unavailable to the client, and the Reader or xArray Gateway will respond to any Impinj extensions with an error. The client sends the custom message to the Reader after the connection is established if it wants to use Impinj extensions. If the connection is lost, the extensions revert to the unavailable state. However, Reader features that are controlled by the earlier use of extensions remain configured through connections, unless otherwise noted.

By sending this message to the Reader, the client acknowledges the ability to process all Impinj extensions. The client must ignore any unrecognized information received from the Reader, including the following:

- Unknown custom messages
- Unknown custom parameters
- Unknown reserved enumeration values in custom parameters
- Use of reserved bits in custom parameters and messages

#### LLRP Dependencies

The IMPINJ_ENABLE_EXTENSIONS message only applies for the duration of the current LLRP connection. If the LLRP connection is broken and re-established, the application must re-issue this command.

#### Definition

**Table 4.2 IMPINJ_ENABLE_EXTENSIONS Message Definition**

| IMPINJ_ENABLE_EXTENSIONS |
|---|
| **Custom Extension Point List**: List of <*Impinj custom parameters*> [optional] |

For more information, see section 6.1.

### 4.2.2 IMPINJ_ENABLE_EXTENSIONS_RESPONSE Message

This custom message is the Reader response to an IMPINJ_ENABLE_EXTENSIONS message. If the Reader is capable of enabling the Impinj extensions, the Reader returns the success code in the LLRPStatus parameter. If there is an error, the Reader returns an appropriate error code.

**LLRP Dependencies**

This custom message has no LLRP dependencies.

**Definition**

**Table 4.3 IMPINJ_ENABLE_EXTENSIONS_RESPONSE Message Definition**

| IMPINJ_ENABLE_EXTENSIONS_RESPONSE |
|---|
| Status: LLRPStatus Parameter<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

For more information, see section 6.1.

### 4.2.3 IMPINJ_SAVE_SETTINGS Message

The IMPINJ_SAVE_SETTINGS custom message instructs the Reader to save the current configuration to persistent storage. The saved parameters then become the power-on and Reader or xArray Gateway reset settings. The specific configuration parameters that are saved to persistent storage are specified using the Boolean fields. These Boolean fields are implemented as a bit-field as shown in section 6.1. Unused reserved bits must be set to zero. Note that there is no way to recall this configuration during runtime. The configuration is only applied after a Reader power-on or reset.

**Speedway Revolution**: The entire reader state is saved to persistent storage. This includes settings from SET_READER_CONFIG, in addition to any configured ROSpecs and AccessSpecs. The current state of *ROSpecs* and *AccessSpecs* is preserved with one exception. The 'Active' *ROSpec* is saved in the 'Inactive' (but enabled) state. This means an *ROSpec* with an 'Immediate' start trigger is saved in the 'Inactive' state, but will then run immediately upon power-on or reset. Similarly, an *ROSpec* with a GPI start trigger will run upon the first GPI transition after power-on or reset. For *AccessSpecs*, the countdown value (if any) is saved as soon as this custom message is received. Automatic update of the persistent configuration during Reader operation is not supported.

**xArray Gateway**: xArray Gateway does not save *ROSpecs* on the IMPINJ_SAVE_SETTINGS command, regardless of the role that is selected.

**LLRP Dependencies**

The configuration of the Reader when the IMPINJ_SAVE_SETTINGS message is received becomes the default configuration for all Reader resets. However, a SET_READER_CONFIG

command with the *ResetToFactoryDefault* flag set will override the persistent settings. The Reader will then initialize with factory settings on subsequent resets until it receives another IMPINJ_SAVE_SETTINGS command.

**Definition**

**Table 4.4 IMPINJ_SAVE_SETTINGS Message Definition**

| IMPINJ_SAVE_SETTINGS |
|---|
| SaveConfiguration: Boolean<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

For more information, see section 6.1.

### 4.2.4 IMPINJ_SAVE_SETTINGS_RESPONSE Message

This custom save-settings message is the response by the Reader to an IMPINJ_SAVE_SETTINGS message. If the Reader was capable of saving the current configuration to persistent storage, the Reader returns the success code in the LLRPStatus parameter. If there is an error, the Reader returns an appropriate error code.

**LLRP Dependencies**

This custom message has no LLRP dependencies.

**Definition**

**Table 4.5 IMPINJ_SAVE_SETTINGS_RESPONSE Message Definition**

| IMPINJ_SAVE_SETTINGS_RESPONSE |
|---|
| Status: LLRPStatus Parameter<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

For more information, see section 6.1.

## 4.3 Custom Parameters

The following subsections describe the custom parameters supported in Octane LLRP. For additional information about using these parameters for xArray Gateway, see section 4.3.43.

### 4.3.1 ImpinjRequestedData Parameter

This custom parameter allows the client to choose specific extensions for inclusion in either a GET_READER_CAPABILITIES_RESPONSE or a GET_READER_CONFIG_RESPONSE message. If the client requests 'All' in the command message and Impinj extensions have been enabled, then all Impinj extensions are included in the response. In order to reduce the response size, the client might request specific response parameters using this extension.

**Note**: This only applies to direct extensions of these two response messages. Custom extensions nested within parameters that are already present in either of these response messages are included provided the Reader has received the IMPINJ_ENABLE_EXTENSIONS message. Although this parameter can appear in either a GET_READER_CAPABILITIES or a GET_READER_CONFIG message, not all ranges for the enumerated *RequestedData* field are valid in both messages.

**LLRP Dependencies**

There are no LLRP dependencies for this custom parameter. The standard LLRP requested data field is processed independently from the custom requested data field, with the exception that 'All' in the standard field also means 'All' in the custom field, if this parameter is omitted and extensions have been enabled.

**Allowable Extension Points**

- GET_READER_CAPABILITIES message (field values 1000 - 1999)
- GET_READER_CONFIG message (field values 2000 - 2999)

**Definition**

**Table 4.6 ImpinjRequestedData Parameter Definitions**

| ***ImpinjRequestedData*** Parameter |
|---|
| RequestedData: Unsigned Integer.<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

**Possible Values**:

| Value | Description |
|---|---|
| 1000 | 'All_Capabilities' |
| 1001 | 'Impinj_Detailed_Version' |
| 1002 | 'Impinj_Frequency_Capabilities' |
| 2000 | 'All_Configuration' |
| 2001 | 'Impinj_Sub_Regulatory_Region' |
| 2003 | 'Impinj_GPI_Debounce_Configuration' |
| 2004 | 'Impinj_Reader_Temperature' |
| 2005 | 'Impinj_Link_Monitor_Configuration' |
| 2006 | 'Impinj_Report_Buffer_Configuration' |
| 2007 | 'Impinj_Access_Spec_Configuration' |
| 2008 | 'Impinj_GPS_NMEA_Sentences' |
| 2009 | 'Impinj_Advanced_GPO_Configuration' |
| 2015 | 'Impinj_Hub_Configuration' |
| All others | Reserved for future use |

For more information about xArray parameters, see section 4.3.43.

### 4.3.2 ImpinjSubRegulatoryRegion Parameter

Use this custom parameter when a particular regulatory region supports multiple operational modes. The Reader validates the *RegulatoryRegion* field against the regulatory regions for which the Reader was manufactured, and only allows compatible regions to be set. Note that when you set the sub-regulatory region by using the SET_READER_CONFIG message, the *ResetToFactoryDefault* field must be set to **true**. This will delete any configured *ROSpecs* and *AccessSpecs*. Failure to set the *ResetToFactoryDefault* field will result in an error. The client should subsequently issue a GET_READER_CAPABILITIES command after it updates the regulatory region, because the change might have affected the advertised Reader capabilities. After the Reader accepts this parameter, it will begin to reload appropriate regulatory settings: this can take several seconds. Applications should plan for an additional delay of several seconds for the SET_READER_CONFIG_RESPONSE.

**Note:** Setting an LLRP *ResetToFactoryDefault* that changes the *RegulatoryRegion* will result in the same behavior as described above.

Not all regulatory regions are supported by each Reader model. Table 4.7 shows supported regions by Reader model.

**Table 4.7 Supported regions by Reader models**

| Region | R220 | R420 | R640 | R680 |
|---|---|---|---|---|
| 0: FCC part 15.247 | √ | √ | √ | √ |
| 1: ETSI EN 300-220 | N/A | N/A | N/A | N/A |
| 2: ETSI EN 302-208 | – | – | – | – |
| 3: Hong Kong 920-925 MHz | √ | √ | √ | √ |
| 4: Taiwan 922-928 MHz | √ | √ | √ | √ |
| 5: Japan 952-954 MHz | N/A | N/A | N/A | N/A |
| 6: Japan 952-955 MHz, 10mW max power | N/A | N/A | N/A | N/A |
| 7: ETSI EN 302-208 (version 1.4.1) | √ | √ | √ | √ |
| 8: Korea 917-921 MHz | √ | √ | √ | √ |
| 9: Malaysia 919-923MHz | √ | √ | √ | √ |
| 10: China 920-925 MHz | √ | √ | √ | √ |
| 11: Japan 952-956 MHz (without LBT) | √ | √ | √ | √ |
| 12: South Africa 915-919 MHz | √ | √ | √ | √ |
| 13: Brazil 902-907/915-928 MHz | √ | √ | √ | √ |
| 14: Thailand 920-925 MHz | √ | √ | √ | √ |
| 15: Singapore 920-925 MHz | √ | √ | √ | √ |
| 16: Australia 920-926 MHz | √ | √ | √ | √ |
| 17: India 865-867 MHz | √ | √ | √ | √ |
| 18: Uruguay 916-928 MHz | √ | √ | √ | √ |
| 19: Vietnam 920-925 MHz | √ | √ | √ | √ |
| 20: Israel 915-917 MHz | √ | √ | √ | √ |
| 21: Philippines 918-920 MHz | √ | √ | √ | √ |
| 22: Canada Post 902-928 MHz | √ | √ | √ | √ |
| 23: Indonesia 923-925 MHz | √ | √ | √ | √ |
| 24: New Zealand 921.5-928MHz | √ | √ | √ | √ |
| 25: Japan 916.7-920.9 MHz | – | √ | – | – |
| 26: Latin America 902-928 MHz | √ | √ | √ | √ |
| 27: Peru 916-928 MHz | √ | √ | √ | √ |

### LLRP Dependencies

When a particular LLRP region supports multiple operational modes, this parameter is required. For example, the LLRP ETSI region might support both with and without LBT. For regions that don't support multiple modes, the Reader will set this parameter automatically, based on the hardware version of the Reader, and the region information specified at manufacturing.

**Allowable Extension Points**

- GET_READER_CONFIG_RESPONSE message
- SET_READER_CONFIG message

**Definition**

**Table 4.8 ImpinjSubRegulatoryRegion Parameter**

| ***ImpinjSubRegulatoryRegion*** Parameter |
|---|
| RegulatoryRegion: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'FCC part 15.247' |
| 1 | 'ETSI EN 300-220' |
| 2 | 'ETSI EN 302-208 (with LBT)'P0F |
| 3 | 'Hong Kong 920-925 MHz' |
| 4 | 'Taiwan 922-928 MHz' |
| 5 | 'Japan 952-954 MHz'P1F - (no longer supported) |
| 6 | 'Japan 952-955 MHz, 10mW max power' - (no longer supported) |
| 7 | 'ETSI EN 302-208 (version 1.4.1)' |
| 8 | 'Korea 917-921 MHz' |
| 9 | 'Malaysia 919-923 MHz' |
| 10 | 'China 920-925 MHz' |
| 11 | 'Japan 952-956 MHz (without LBT)' |
| 12 | 'South Africa 915-919 MHz' |
| 13 | 'Brazil 902-907 and 915-928 MHz' |
| 14 | 'Thailand 920-925 MHz' |
| 15 | 'Singapore 920-925 MHz' |
| 16 | 'Australia 920-926 MHz' |
| 17 | 'India 865-867 MHz' |
| 18 | 'Uruguay 916-928 MHz' |
| 19 | 'Vietnam 920-925 MHz' |
| 20 | 'Israel 915-917 MHz' |
| 21 | 'Philippines 918-920 MHz' |
| 22 | 'Canada Post 902-928 MHz' |

| Value | Description |
|---|---|
| 23 | 'Indonesia 923-925 MHz' |
| 24 | 'New Zealand 921.5-928 MHz' |
| 25 | 'Japan 916.7-920.9 MHz' |
| 26 | 'Latin America 902-928 MHz' |
| 27 | 'Peru 916-928 MHz' |
| 28-65535 | Reserved for future use |

### 4.3.3 ImpinjInventorySearchMode Parameter

Specify the Impinj-specific inventory search mode used by a particular antenna using this custom parameter. The inventory search mode may be configured as either part of the default Reader configuration (via a SET_READER_CONFIG message), or as part of individual AISpecs within a ROSpec (via an ADD_ROSPEC message). For any AISpec, each enabled antenna must be configured to use the same search algorithm. Mismatched antenna settings will result in an error reported by the Reader.

**LLRP Dependencies**

Impinj Readers implement state unaware singulation and therefore the Client does not control how the Reader attempts to singulate tags. This parameter provides a high-level control over the search algorithm and consequently does not interfere with any of the standard LLRP settings. When the InventorySearchMode is set to zero, the Reader will pick the inventory search mode that provides the most consistent performance for the session and timing parameters provided by LLRP.

**Allowable Extension Points**

- *C1G2InventoryCommand* parameter

**Definition**

**Table 4.9 ImpinjInventorySearchMode Parameter**

| ImpinjInventorySearchMode **Parameter** |
|---|
| InventorySearchMode: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Reader selected mode' (default) |
| 1 | 'Single target inventory' High tag count, high-throughput applications where a reduction in repeated tag observation is acceptable. |
| 2 | 'Dual target inventory' Low to medium tag count or low-throughput applications where repeated tag observation is desirable. |
| 3 | 'Single target inventory' (with suppressed duplicate observations). Maximum tag count, high-throughput applications where a single observation of each tag is acceptable. This search mode suppresses repeated observations for extended periods of time while tags are energized. (Available only for use with Session 1.) |
| 4-65535 | Reserved for future use |

### 4.3.4 ImpinjFixedFrequencyList Parameter

Use this custom parameter to allow the Reader to make intelligent decisions about which channel to use in fixed frequency or intelligent hopping regulatory regions. The *FixedFrequencyMode* field determines how the Reader will select the active channel. When set to **zero** (disabled), the Reader ignores this parameter and instead uses the frequency information in the LLRP *RFTransmitter* parameter. When set to **1**, the Reader chooses the active channel automatically , based on the rules of the regulatory region. When set to **2**, the Reader chooses the active channel from a configurable list of channel indices based on the *ImpinjFrequencyCapabilities* parameter advertised in the Reader's capabilities. The maximum number of channels allowed in the configurable list is shown in Table 4.10. A repeated channel index means the Reader will test the channel multiple times. When *FixedFrequencyMode* is set to zero or one, the *ChannelList* array is ignored.

This parameter can only be added in an operational mode, such as 'not disabled', when the Reader is operating in a fixed frequency or intelligent hopping regulatory region. The parameter contents must be consistent across all enabled antennas in an AISpec.

**Table 4.10 ImpinjFixedFrequencyList Regulatory Information**

| Region | Available FixedFrequencyMode | Maximum ChannelList Size |
|---|---|---|
| 0: FCC part 15.247 | 0 | N/A |
| 1: ETSI EN 300-220 | N/A | N/A |
| 2: ETSI EN 302-208 | See region 7 | See region 7 |
| 3: Hong Kong 920-925 MHz | 0 | N/A |
| 4: Taiwan 922-928 MHz | 0 | N/A |

| Region | Available FixedFrequencyMode | Maximum ChannelList Size |
|---|---|---|
| 5: Japan 952-954 MHz | N/A | N/A |
| 6: Japan 952-955 MHz, 10mW max power | N/A | N/A |
| 7: ETSI EN 302-208 (version 1.4.1) | 0, 1, 2 | 4 |
| 8: Korea 917-921 MHz | 0 | N/A |
| 9: Malaysia 919-923MHz | 0 | N/A |
| 10: China 920-925 MHz | 0, 1, 2 | 16 |
| 11: Japan 952-956 MHz (without LBT) | 0, 1, 2 | 4 |
| 12: South Africa 915-919 MHz | 0 | N/A |
| 13: Brazil 902-907/915-928 MHz | 0 | N/A |
| 14: Thailand 920-925 MHz | 0 | N/A |
| 15: Singapore 920-925 MHz | 0 | N/A |
| 16: Australia 920-926 MHz | 0 | N/A |
| 17: India 865-867 MHz | 0, 1, 2 | 4 |
| 18: Uruguay 916-928 MHz | 0 | N/A |
| 19: Vietnam 920-925 MHz | 0 | N/A |
| 20: Israel 915-917 MHz | 0 | N/A |
| 21: Philippines 918-920 MHz | 0, 1, 2 | 4 |
| 22: Canada Post 902-928 MHz | 0 | N/A |
| 23: Indonesia 923-925 MHz | 0 | N/A |
| 24: New Zealand 921.5-928MHz | 0 | N/A |
| 25: Japan 916.7-920.9 MHz | 0, 1, 2 | 4 |
| 26: Latin America 902-928 MHz | 0 | N/A |
| 27: Peru 916-928 MHz | 0 | N/A |

### LLRP Dependencies

When present and enabled, this parameter overrides the *ChannelIndex* field of the *RFTransmitter* parameter. The Reader will always return the last value set in the *ChannelIndex* field if queried, but if a client sets this custom parameter, that value must be ignored.

### Allowable Extension Points

- *C1G2InventoryCommand* parameter

**Definition**

**Table 4.11 ImpinjFixedFrequencyList Parameter**

| ***ImpinjFixedFrequencyList*** Parameter |
|---|
| FixedFrequencyMode: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Disabled' (default) |
| 1 | 'Reader will choose the channel from those allowed in the current regulatory region' |
| 2 | 'Reader will choose the channel from the channel indices provided in the ChannelList field' |
| 3-65535 | Reserved for future use |

**Note:** *ChannelList*: Short Array. An array of indices into the Reader's *ImpinjFrequencyCapabilities* that can be used.

### 4.3.5 ImpinjFrequencyCapabilities Parameter

The frequency capabilities custom parameter is included in the Reader's capabilities and carries each frequency supported by the Reader (see section 4.2.6 for more information). The *FrequencyList* field is a one-based array of frequencies in kHz.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- GET_READER_CAPABILITIES_RESPONSE

**Definition**

**Table 4.12 ImpinjFrequencyCapabilities Parameter**

| ***ImpinjFrequencyCapabilities*** Parameter |
|---|
| FrequencyList: Unsigned Integer Array. Frequency in kHz.<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

### 4.3.6 ImpinjReducedPowerFrequencyList Parameter

This custom parameter is used to allow the Reader to apply a reduced power to specific channels listed in the *ChannelList*, when they are operating in the FCC regulatory region. The *ReducedPowerMode* field determines how the Reader will interpret the channels listed. When set to zero (disabled), the Reader ignores this parameter. When set to **1**, during inventory and access the Reader will apply a reduced power level to the list of channel indices derived from the *FrequencyList*. This list is supplied by the *ImpinjFrequencyCapabilities* parameter advertised in the Reader's capabilities (see section 4.3.5). The reduced power level is not configurable, and defaults to the lowest device power possible. The maximum number of channels allowed in the configurable list are specified in Table 4.13. When *ReducedPowerMode* is **1**, the *ChannelList* must contain at least two channels and no channel can be repeated. This parameter is valid only when the Reader is operating in the FCC regulatory region as shown in Table 4.13). The parameter contents must be consistent across all enabled antennas in an AISpec.

**Table 4.13 ImpinjReducedPowerFrequencyList Regulatory Information**

| Region | Available ReducedPowerMode | Maximum ChannelList Size |
|---|---|---|
| 0: FCC part 15.247 | 0, 1 | 16 |
| 1. ETSI EN 300-220 | N/A | N/A |
| 2: ETSI EN 302-208 | See region 7 | See region 7 |
| 3: Hong Kong 920-925 MHz | 0 | N/A |
| 4: Taiwan 922-928 MHz | 0 | N/A |
| 5: Japan 952-954 MHz | N/A | N/A |
| 6: Japan 952-955 MHz, 10mW max power | N/A | N/A |
| 7: ETSI EN 302-208 (version 1.4.1) | 0 | N/A |
| 8: Korea 917-921 MHz | 0 | N/A |
| 9: Malaysia 919-923MHz | 0 | N/A |

| Region | Available ReducedPowerMode | Maximum ChannelList Size |
|---|---|---|
| 10: China 920-925 MHz | 0 | N/A |
| 11: Japan 952-956 MHz (without LBT) | 0 | N/A |
| 12: South Africa 915-919 MHz | 0 | N/A |
| 13: Brazil 902-907/915-928 MHz | 0 | N/A |
| 14: Thailand 920-925 MHz | 0 | N/A |
| 15: Singapore 920-925 MHz | 0 | N/A |
| 16: Australia 920-926 MHz | 0 | N/A |
| 17: India 865-867 MHz | 0 | N/A |
| 18: Uruguay 916-928 MHz | 0 | N/A |
| 19: Vietnam 920-925 MHz | 0 | N/A |
| 20: Israel 915-917 MHz | 0 | N/A |
| 21: Philippines 918-920 MHz | 0 | N/A |
| 22: Canada Post 902-928 MHz | 0 | N/A |
| 23: Indonesia 923-925 MHz | 0 | N/A |
| 24: New Zealand 921.5-928MHz | 0 | N/A |
| 25: Japan 916.7-920.9 MHz | 0 | N/A |
| 26: Latin America 902-928 MHz | 0 | N/A |
| 27: Peru 916-928 MHz | 0 | N/A |

### LLRP Dependencies

This custom parameter has no LLRP dependencies.

### Allowable Extension Points

- *C1G2InventoryCommand* parameter

### Definition

**Table 4.14 ImpinjReducedPowerFrequencyList Parameter**

| ***ImpinjReducedPowerFrequencyList*** Parameter |
|---|
| ReducedPowerMode: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Disabled' (default) |
| 1 | 'Reader applies the reduced power level to the Channels specified in the ChannelList' |
| 2-65535 | 'Reserved for future use' |

**Note:** *ChannelList*: Unsigned Short Array. A maximum of sixteen one-based indices into the Reader's :under:FrequencyList' as advertised in the *ImpinjFrequencyCapabilities* parameter to apply the reduced power during inventory and access.

### 4.3.7 ImpinjLowDutyCycle Parameter

Use this custom parameter to provide additional control of the RF duty cycle of the Reader beyond the control already provided by the *TagTransitTime* field in the LLRP *C1G2SingulationControl* parameter. During inventory, if the Reader detects zero tags in the field-of-view (a definition that is model-specific, as described in this section), *EmptyFieldTimeout* specifies in milliseconds the amount of time the Reader will wait before entering low duty cycle mode. In this low duty cycle mode, the Reader will rescan the FOV every *FieldPingInterval* milliseconds, checking for tags. When a tag is detected, the full duty cycle will resume. The Reader exits the low duty cycle mode at the start of each AISpec, and restarts the empty field timers.

For regulatory region compliance, low duty cycle operation will occur in some regions, whether low duty cycle operation parameters are specified or not. For such regions, if valid low duty cycle operation parameter values are specified, the Reader might choose to adjust the specified values in order to maintain regulatory region compliance. This parameter is invalid in regions that use LBT, as shown in Table 4.15.

The FOV is defined as the tags visible by a single antenna, independent of the other antennas enabled in the current AISpec. Thus, each antenna manages its own FOV, empty field timer, and field ping timer. This means that the low duty cycle settings can be configured independently from other antennas that are enabled in the current AISpec. The only requirement is that, if one antenna in the AISpec uses the *ImpinjLowDutyCycle* parameter, all antennas must use the extension. However, the timer values can vary between enabled antennas.

As an example, assume that, for one of the antennas in the current AISpec, *EmptyFieldTimeout* is set to 500 milliseconds and *FieldPingInterval* is set to 200 milliseconds . After that antenna detects zero tags in the field-of-view, that antenna's empty field timer is started. If that antenna subsequently detects tags, the timer stops. If that antenna detects zero tags for 500 milliseconds, a timeout occurs and the antenna enters low duty cycle mode. During this mode, the antenna will switch on briefly every 200 milliseconds to check for tags in its FOV. While this is all occurring, the same algorithm is running on each of the other enabled antennas independently.

**Table 4.15 ImpinjLowDutyCycle Regulatory Information**

| Region | Available LowDutyCycleMode |
|---|---|
| 0: FCC part 15.247 | 0, 1 |
| 1: ETSI EN 300-220 | N/A |
| 2: ETSI EN 302-208 | See region 7 |
| 3: Hong Kong 920-925 MHz | 0, 1 |
| 4: Taiwan 922-928 MHz | 0, 1 |
| 5: Japan 952-954 MHz | N/A |
| 6: Japan 952-955 MHz, 10mW max power | N/A |
| 7: ETSI EN 302-208 (version 1.4.1) | 0, 1 |
| 8: Korea 917-921 MHz | 0, 1 |
| 9: Malaysia 919-923MHz | 0, 1 |
| 10: China 920-925 MHz | 0, 1 |
| 11: Japan 952-956 MHz (without LBT) | 0, 1 |
| 12: South Africa 915-919 MHz | 0, 1 |
| 13: Brazil 902-907/915-928 MHz | 0, 1 |
| 14: Thailand 920-925 MHz | 0, 1 |
| 15: Singapore 920-925 MHz | 0, 1 |
| 16: Australia 920-926 MHz | 0, 1 |
| 17: India 865-867 MHz | 0, 1 |
| 18: Uruguay 916-928 MHz | 0, 1 |
| 19: Vietnam 920-925 MHz | 0, 1 |
| 20: Israel 915-917 MHz | 0, 1 |
| 21: Philippines 918-920 MHz | 0, 1 |
| 22: Canada Post 902-928 MHz | 0, 1 |
| 23: Indonesia 923-925 MHz | 0, 1 |
| 24: New Zealand 921.5-928MHz | 0, 1 |
| 25: Japan 916.7-920.9 MHz | 0, 1 |
| 26: Latin America 902-928 MHz | 0, 1 |
| 27: Peru 916-928 MHz | 0, 1 |

**xArray**

This custom parameter is not supported by xArray. This parameter is not reported and any command that uses this parameter will be rejected.

**LLRP Dependencies**

If present and enabled, this parameter overrides the *TagTransitTime* field in the LLRP *C1G2SingulationControl* parameter. The Reader always returns the last value that was set in the *TagTransitTime* field if queried. However, if a client has set this custom parameter, that value must be ignored.

**Allowable Extension Points**

- *C1G2InventoryCommand* parameter

**Definition**

**Table 4.16 ImpinjLowDutyCycle Parameter**

| ***ImpinjLowDutyCycle*** Parameter |
|---|
| LowDutyCycleMode: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Disabled' (default) |
| 1 | 'Enabled' |
| 2-65535 | Reserved for future use |

**Note:** *EmptyFieldTimeout*: Unsigned Short Integer. The time in milliseconds the Reader will wait, having detected no tags on all enabled antennas, before switching to low duty cycle mode.

**Note:** *FieldPingInterval*: Unsigned Short Integer. The time in milliseconds before the Reader switches on the transmitter to search for tags in the field during low duty cycle mode.

### 4.3.8 ImpinjDetailedVersion Parameter

Use this custom parameter to provide detailed information about the individual components running on the Reader. The primary platform version is available in the *ReaderFirmwareVersion*' field in the *GeneralDeviceCapabilities* parameter of the Reader's capabilities. However, there are

sub-components of the Reader that contain independent version information that is unavailable by using the primary platform version. This custom parameter provides the detailed information. The meaning behind each field varies by Reader model. Table 4-17 provides the translation for each model type.

**Table 4.17 ImpinjDetailedVersion Field Descriptions**

| Field | Speedway Revolution |
|---|---|
| *ModelName* | The model name of the reader. |
| *SerialNumber* | The serial number of the reader. |
| *SoftwareVersion* | The primary platform firmware version (SOP). Same as *ReaderFirmwareVersion*. |
| *FirmwareVersion* | The firmware version of the Command Sequencer component. |
| *FPGAVersion* | The firmware version of the FPGA component. |
| *PCBAVersion* | The hardware version of the PCBA component. |
| *ImpinjHubVersions* | Hardware, firmware versions and and serial numbers of attached Antenna Hubs. |
| *ImpinjArrayVersion* | Array specific hardware and version information that is considered additional to ImpinjDetailedVersion information. |

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- GET_READER_CAPABILITIES_RESPONSE message

**Definition**

**Table 4.18 ImpinjDetailedVersion Parameter**

| ***ImpinjDetailedVersion*** Parameter |
|---|
| *ModelName*: UTF-8 String<br>*SerialNumber*: UTF-8 String<br>*SoftwareVersion*: UTF-8 String<br>*FirmwareVersion*: UTF-8 String<br>*FPGAVersion*: UTF-8 String<br>*PCBAVersion*: UTF-8 String<br>*ImpinjHubVersions*: Custom Parameter<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

**Table 4.19 ImpinjHubVersions Parameter**

| ***ImpinjHubVersions*** Parameter |
|---|
| List of *ImpinjArrayVersion* parameters<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

**Table 4.20 ImpinjArrayVersion Parameter**

| ***ImpinjArrayVersion*** Parameter |
|---|
| *SerialNumber*: UTF-8 String<br>*FirmwareVersion*: UTF-8 String<br>*PCBAVersion*: UTF-8 String<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

### 4.3.9 ImpinjGPIDebounceConfiguration Parameter

Use this custom parameter to control the GPI debounce timing. The *GPIPortNum* field is the 1-based GPI number, identical to *GPIPortNum* in the *GPIPortCurrentState* LLRP parameter. Once a transition is detected, whether rising or falling, subsequent transitions are ignored for

*GPIDebounceTimerMSec* milliseconds. This timer value must be a multiple of 10 milliseconds. Setting *GPIDebounceTimerMSec* to zero effectively disables debounce. The GPI debounce timer affects triggered ROSpecs and GPI event reporting.

**xArray**

This custom parameter is not supported by xArray. This parameter will not be reported and any command that uses this parameter will be rejected.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- GET_READER_CONFIG_RESPONSE message
- SET_READER_CONFIG message

**Definition**

**Table 4.21 ImpinjGPIDebounceConfiguration Parameter**

| ***ImpinjGPIDebounceConfiguration*** Parameter |
|---|
| *GPIPortNum*: Unsigned Short Integer.<br>*Unsigned Integer*: The debounce duration in milliseconds. Must be a multiple of 10 milliseconds. Zero turns off the debounce algorithm for this GPI.<br>**Cust0m Extension Point List**: List of <Impinj Custom Parameters> [optional] |

### 4.3.10 ImpinjAdvancedGPOConfiguration Parameter

Use this custom parameter to control the advanced GPO feature of Speedway Revolution readers. When set to **Normal** (default) the GPO is set via the regular LLRP SET_READER_CONFIG message. When set to **Pulsed**, the GPO changes state based on the SET_READER_CONFIG message, and will change to the opposite state after *GPOPulseDurationMSec* milliseconds. When set to 'Reader_Operational_Status', 'LLRP_Connection_Status', 'Reader_Inventory_Status', 'Network_Connection_Status', or 'Reader_Inventory_Tags_Status', the GPO status acts like a Boolean value. When high (**true**, **1**), the corresponding status is **true**, which means that the reader is operating, has a LLRP connection, is inventorying, has a network connection, or tags are being singulated (respectively). When low (**false**, **0**), the opposite is the case. The GPO might

lag the actual internal status. Notably the worst case delay on the 'Network_Connection_Status' can be up to 17 seconds.

**xArray**

This custom parameter is not supported by xArray. This parameter will not be reported and any command that uses this parameter will be rejected.

**LLRP Dependencies**

Whenever a GPO has been associated with a specific reader status, it cannot be set via the normal LLRP protocol. If a SET_READER_CONFIG message is received that attempts to change the state of a GPO that is associated with a specific reader status, the message will be rejected by the Reader.

**Allowable Extension Points**

- GET_READER_CONFIG_RESPONSE message
- SET_READER_CONFIG message

**Definition**

**Table 4.22 ImpinjAdvancedGPOConfiguration Parameter**

| ***ImpinjAdvancedGPOConfiguration*** Parameter |
|---|
| *GPOPortNum*: Unsigned Short Integer.<br>*GPOMode*: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <Impinj Custom Parameters> [optional] |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Normal' (default) |
| 1 | 'Pulsed' |
| 2 | 'Reader_Operational_Status' |
| 3 | 'LLRP_Connection_Status' |
| 4 | 'Reader_Inventory_Status' |
| 5 | 'Network_Connection_Status' |
| 6 | 'Reader_Inventory_Tags_Status' |
| 7-65535 | Reserved for future use |

**Note:** *GPOPulseDurationMSec*: Unsigned Integer. The duration of the GPO pulse. This field is only valid when *GPOMode* is set to **Pulsed**. When GPOMode is set to **Pulsed**, this value must be non-zero. The duration is specified in milliseconds.

### 4.3.11 ImpinjReaderTemperature Parameter

Use this custom parameter to report the current temperature of the Reader in degrees Celsius. The temperature that is reported is the internal temperature of the Reader or xArray Gateway, not the ambient temperature of the Reader surroundings. The temperature is accurate to within 2 |deg| C across all operating temperatures.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- GET_READER_CONFIG_RESPONSE message

**Definition**

**Table 4.23 ImpinjReaderTemperature Parameter**

| ***ImpinjReaderTemperature*** Parameter |
|---|
| *Temperature*: Signed Short Integer. The current temperature in degrees Celsius.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional] |

### 4.3.12 ImpinjLinkMonitorConfiguration Parameter

Use this custom parameter to configure the Reader to monitor LLRP link health by using KEEPALIVE and KEEPALIVE_ACK messages. When disabled or unsupported (see Table 4.1), the Reader ignores KEEPALIVE_ACK messages entirely. When this parameter is enabled, if the Reader fails to receive *LinkDownThreshold* consecutive KEEPALIVE_ACK messages from the client, the Reader will close the current connection. Note that this parameter must be configured in conjunction with the *KeepaliveSpec* LLRP parameter in the standard LLRP configuration. The frequency with which the Reader is configured to send KEEPALIVE messages, along with the threshold set in this parameter, determines how long the Reader will tolerate missing KEEPALIVE_ACK messages. The Reader or xArray Gateway uses the LLRP *MessageID*

field to correlate KEEPALIVE and KEEPALIVE_ACK messages. Clients must send the same *MessageID* when responding to Reader KEEPALIVE requests.

**LLRP Dependencies**

This custom parameter must be set in conjunction with the LLRP *KeepaliveSpec* parameter. If you setting this parameter alone without configuring a Periodic KeepaliveSpec, it has no effect.

**Allowable Extension Points**

- GET_READER_CONFIG_RESPONSE message
- SET_READER_CONFIG message

**Definition**

**Table 4.24 ImpinjLinkMonitorConfiguration Parameter**

| ***ImpinjLinkMonitorConfiguration*** Parameter |
|---|
| *LinkMonitorMode*: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <Impinj Custom Parameters> [optional |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Disabled' (default) |
| 1 | 'Enabled' |
| 2-65535 | Reserved for future use |

**Note:** *LinkDownThreshold*: Unsigned Short Integer. The number of consecutive KEEPALIVE_ACK response messages that were not received before the Reader closed the current connection.

### 4.3.13 ImpinjReportBufferConfiguration Parameter

Use this custom parameter to configure how the Reader buffers asynchronous reports sent to the client. In **Normal** mode, the Reader buffers RO_ACCESS_REPORT messages internally for an optimal time period before transmission over the network. Response messages, KEEPALIVE messages, and READER_EVENT_NOTIFICATION messages are not affected, and are sent immediately. In **Low_Latency** mode, the Reader sends RO_ACCESS_REPORT messages as soon as they are available. In general, the default mode is well suited to most applications. Applications

that require immediate access to inventory reports may require **Low_Latency** mode, but users should first evaluate network and system load.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- GET_READER_CONFIG_RESPONSE message
- SET_READER_CONFIG message

**Definition**

**Table 4.25 ImpinjReportBufferConfiguration Parameter**

| ***ImpinjReportBufferConfiguration*** Parameter |
|---|
| *ReportBufferMode*: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <Impinj Custom Parameters> [optional |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Normal' (default) |
| 1 | 'Low_Latency' |
| 2-65535 | Reserved for future use |

### 4.3.14 ImpinjAccessSpecConfiguration Parameter

Use this custom parameter to allow additional control over how the Reader executes *AccessSpecs*. This parameter does not contain any specific controls, but it encapsulates individual parameters that do. Each parameter that it contains is optional, which allows for maximum flexibility for client implementations.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies, although the parameters that it contains might. Reference the individual parameters for information about how they affect LLRP behavior.

**Allowable Extension Points**

- GET_READER_CONFIG_RESPONSE message

- SET_READER_CONFIG message
- *AccessSpec* parameter

**Definition**

**Table 4.26 ImpinjAccessSpecConfiguration Parameter**

| ***ImpinjAccessSpecConfiguration*** Parameter |
|---|
| *ImpinjBlockWriteWordCount*: <*ImpinjBlockWriteWordCount* parameter> [optional] |
| *ImpinjOpSpecRetryCount*: <*ImpinjOpSpecRetryCount* parameter> [optional] |
| **Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional] |

### 4.3.15 ImpinjBlockWriteWordCount Parameter

Use this parameter to configure the number of words sent at one time to a tag, when processing a *C1G2BlockWrite* OpSpec custom parameter. The LLRP *C1G2BlockWrite* parameter has a word vector that contains the data to be written to a tag. Internally, the Reader breaks this vector up into individual C1G2 BlockWrite commands. This parameter determines the number of words sent via each BlockWrite command. Note that it is the user's responsibility to ensure that the tag population supports the BlockWrite word count that is configured via this parameter. BlockWrite commands to tags that do not support the configured word count will fail. The Reader or xArray Gateway automatically aligns C1G2BlockWrite commands to appropriate boundaries and accounts for odd data lengths. The default word count is one.

**LLRP Dependencies**

This custom parameter determines the number of words sent at a time over the C1G2 air interface when processing a LLRP *C1G2BlockWrite* parameter.

**Allowable Extension Points**

- None included in *ImpinjAccessSpecConfiguration* parameter.

**Definition**

**Table 4.27 ImpinjBlockWriteWordCount Parameter**

| ***ImpinjBlockWriteWordCount*** Parameter |
|---|
| *WordCount*: Unsigned Short Integer. Allowable range is 1-2.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional] |

### 4.3.16 ImpinjOpSpecRetryCount Parameter

Yse this custom parameter to configure the number of times an OpSpec operation will be automatically retried by the Reader before failure is declared . The Reader intelligently chooses which types of failures to retry. For example, the Reader will not retry if the tag indicates that the operation failed due to a memory locked or memory overrun, which are operations that have no chance of succeeding. However, if the operation failed due to transient errors, such as CRC errors due to interference, the Reader will automatically retry *RetryCount* attempts before failure is declared. LLRP dictates that OpSpec failure be declared once a single operation has failed, therefore the default *RetryCount* is 0.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- None included in the *ImpinjAccessSpecConfiguration* parameter.

**Definition**

**Table 4.28 ImpinjOpSpecRetryCount Parameter**

| ***ImpinjOpSpecRetryCount*** Parameter |
|---|
| *RetryCount*: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional |

**Possible Values**:

| Value | Description |
|---|---|
| 0-3 | The number of times each operation is retried. |

### 4.3.17 ImpinjBlockPermalock Parameter

This OpSpec custom parameter configures the *C1G2 BlockPermalock* status of a particular memory bank from a tag. The *AccessPassword* field is the password that is required to move the tag into the secured state, if needed.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- *AccessCommandOpSpec* choice parameter.

**Definition**

**Table 4.29 ImpinjBlockPermalock Parameter**

| ***ImpinjBlockPermalock*** Parameter |
|---|
| *OpSpecID*: Unsigned Short Integer.<br>*AccessPassword* : Unsigned Integer.<br>*MB*: Integer. The memory bank on which to perform the *BlockPermalock*.<br>*Possible Values*: 0-3<br>*BlockPointer*: Unsigned Short Integer. Specifies the starting address for *BlockMask*‘ in units of 16 blocks.<br>*BlockMask*: Unsigned Short Integer Array. Specifies the blocks to lock, starting at *BlockPointer* and ending ((16*(*BlockMask* array length)) - 1) blocks later.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional] |

### 4.3.18 ImpinjBlockPermalockOpSpecResult Parameter

This custom parameter is the result of an *ImpinjBlockPermalock* OpSpec.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- *AccessCommandOpSpecResult* choice parameter.

**Definition**

**Table 4.30 ImpinjBlockPermalockOpSpecResult Parameter**

| ***ImpinjBlockPermalockOpSpecResult*** Parameter |
|---|
| *OpSpecID*: Unsigned Short Integer.<br>Result: Integer.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Success' |
| 1 | 'Insufficient power to perform block permalock operation' |
| 2 | 'Non-specific tag error' |
| 3 | 'No response from tag' |
| 4 | 'Non-specific Reader error' |
| 5 | 'Incorrect password error' |
| 6 | 'Tag memory overrun error' |

### 4.3.19 ImpinjGetBlockPermalockStatus Parameter

Use this custom parameter to retrieve the OpSpec *C1G2 BlockPermalock* status of a particular memory bank from a tag. The *AccessPassword* field is the password that is required to move the tag into the secured state, if needed.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- *AccessCommandOpSpec* choice parameter.

**Definition**

**Table 4.31 ImpinjGetBlockPermalockStatus Parameter**

| ***ImpinjGetBlockPermalockStatus*** Parameter |
|---|
| *OpSpecID*: Unsigned Short Integer.<br>*AccessPassword*: Unsigned Integer.<br>*MB*: Integer. The memory bank on which to retrieve the BlockPermalock status.<br>*Possible Values*: 0-3<br>*BlockPointer*: Unsigned Short Integer. Specifies the starting address to retrieve in units of 16 blocks.<br>*BlockRange*: Unsigned Short Integer. Specifies the range of blocks to retrieve, starting at *BlockPointer* and ending ((16 * *BlockRange*) - 1) blocks later.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional] |

### 4.3.20 ImpinjGetBlockPermalockStatusOpSpecResult Parameter

This custom parameter is the result of an *ImpinjGetBlockPermalockStatus* OpSpec.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- *AccessCommandOpSpecResult* choice parameter.

**Definition**

**Table 4.32 ImpinjGetBlockPermalockStatusOpSpecResult Parameter**

| ***ImpinjGetBlockPermalockStatusOpSpecResult*** Parameter |
|---|
| *OpSpecID*: Unsigned Short Integer.<br>*PermalockStatus*: Unsigned Short Integer Array. Specifies the Permalock status of each block requested.<br>Result: Integer.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Success' |
| 1 | 'Non-specific tag error' |
| 2 | 'No response from tag' |
| 3 | 'Non-specific Reader error' |
| 4 | 'Incorrect password error' |
| 5 | 'Tag memory overrun error' |

### 4.3.21 ImpinjSetQTConfig Parameter

Use this custom parameter to set the OpSpec for the QT Technology |trade| configuration on Impinj Monza 4QT tags. For more information about the meaning of the fields in this parameter and the use cases for this technology, see the Impinj Monza 4QT datasheet. Some tags might not be reported when you use Serialized TID reporting and Monza4-QT tags with both public and short range modes. For more information, see section 5.1 **Speedway Revolution Serialized TID Reporting and Monza4 Tags**.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- AccessCommandOpSpec choice parameter.

**Definition**

**Table 4.33 ImpinjSetQTConfig Parameter**

| ***ImpinjSetQTConfig*** Parameter |
|---|
| *OpSpecID*: Unsigned Short Integer.<br>*AccessPassword*: Unsigned Integer.<br>*DataProfile*: Integer. Determines which data profile is exposed by the tag. |

**Possible Values**:

| Value | Description |
|---|---|
| 1 | 'Private. The tag exposes its private data profile.' |
| 2 | 'Public. The tag exposes its public data profile.' |

| Value | Description |
|---|---|
| 0,3-255 | 'Reserved for future use' |

*AccessRange*': Integer. Determines the range at which the tag may be accessed (Read, Write, Lock, etc.). The range at which the tag is inventoried is not affected.

**Possible Values**:

| Value | Description |
|---|---|
| 1 | 'Normal_Range.' The tag responds to access operations at the maximum range supported by the environment. |
| 2 | Short_Range. The tag only responds to access operations from a short range. |
| 0,3-255 | Reserved for future use |

*Persistence*: Integer. Determines how long the changes made to the QT configuration with this OpSpec remain in effect.

**Possible Values**:

| Value | Description |
|---|---|
| 1 | 'Temporary.' The changes made by this command only last until the tag is powered down, at which time the previous configuration is restored. |
| 2 | 'Permanent.' The changes made by this command are stored permanently to nonvolatile memory. |
| 0,3-255 | Reserved for future use |

**Note: Custom Extension Point List**: List of <Impinj Custom Parameters> [optional]

### 4.3.22 ImpinjSetQTConfigOpSpecResult Parameter

This custom parameter is the result of an *ImpinjSetQTConfig* OpSpec.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- *AccessCommandOpSpecResult* choice parameter.

**Definition**

**Table 4.34 ImpinjSetQTConfigOpSpecResult Parameter**

| ***ImpinjSetQTConfigOpSpecResult*** Parameter |
|---|
| *OpSpecID*: Unsigned Short Integer.<br>Result: Integer.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Success' |
| 1 | 'Insufficient power to perform QT write operation' |
| 2 | 'Non-specific tag error' |
| 3 | 'No response from tag' |
| 4 | 'Non-specific Reader error' |
| 5 | 'Incorrect password error' |

### 4.3.23 ImpinjGetQTConfig Parameter

Use this custom parameter to retrieve the OpSpec QT Technology |trade| configuration on Impinj Monza 4QT tags . For more information about the meaning of the fields within this parameter, and the use cases for this technology, refer to the Impinj Monza 4QT datasheet.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- *AccessCommandOpSpec* choice parameter.

**Definition**

**Table 4.35 ImpinjGetQTConfig Parameter**

| ***ImpinjGetQTConfig*** Parameter |
|---|
| *OpSpecID*: Unsigned Short Integer.<br>*AccessPassword*: Unsigned Integer.<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

### 4.3.24 ImpinjGetQTConfigOpSpecResult Parameter

This custom parameter is the result of an *ImpinjGetQTConfig* OpSpec.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- AccessCommandOpSpecResult choice parameter.

**Definition**

**Table 4.36 ImpinjGetQTConfigOpSpecResult Parameter**

| ***ImpinjGetQTConfigOpSpecResult*** Parameter |
|---|
| *OpSpecID*: Unsigned Short Integer.<br>Result: Integer. |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Success' |
| 1 | 'Non-specific tag error' |
| 2 | 'No response from tag' |
| 3 | 'Non-specific Reader error' |
| 4 | 'Incorrect password error' |

| |
|---|
| *DataProfile*: Integer. Determines which data profile is exposed by the tag. |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Unknown' |
| 1 | 'Private' The tag exposes its private data profile. |
| 2 | 'Public' The tag exposes its public data profile. |
| 3-255 | Reserved for future use |

*AccessRange*: Integer. Determines the range at which the tag may be accessed (Read, Write, Lock, etc.). The range at which the tag is inventoried is not affected.

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Unknown'. |
| 1 | 'Normal_Range'. The tag responds to access operations at the maximum range supported by the environment. |
| 2 | 'Short_Range'. The tag only responds to access operations from a short range. |
| 3-255 | Reserved for future use |

**Note: Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional]

### 4.3.25 ImpinjTagReportContentSelector Parameter

Use this custom parameter to configure additional parameters that are reported via the *TagReportData* parameter. Each optional parameter individually enables or configures a particular feature. Note that, because of how the *ROReportSpec* parameter is handled (described in section 3.1.15), if the optional parameter used to control a particular feature is absent, the feature is considered disabled. See the documentation for the actual parameters for full feature descriptions.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- *ROReportSpec* parameter.

**Definition**

**Table 4.37 ImpinjTagReportContentSelector Parameter**

| ***ImpinjTagReportContentSelector*** Parameter |
|---|
| *ImpinjEnableSerializedTID*: <ImpinjEnableSerializedTID parameter> [optional]<br>*ImpinjEnableRFPhaseAngle*: <ImpinjEnableRFPhaseAngle parameter> [optional]<br>*ImpinjEnablePeakRSSI*: <ImpinjEnablePeakRSSI parameter> [optional]<br>*ImpinjEnableGPSCoordinates*: <ImpinjEnableGPSCoordinates parameter> [optional]<br>*ImpinjEnableOptimizedRead*: <ImpinjEnableOptimizedRead parameter> [optional]<br>**Custom Extension Point List**: List of <Impinj custom parameter> [optional] |

### 4.3.26 ImpinjEnableSerializedTID Parameter

Use this custom parameter to configure the Impinj Serialized TID feature. For tags that support this feature, when it is enabled, the *TagReportData* in the RO_ACCESS_REPORT will contain an *ImpinjSerializedTID* parameter that reports the tag TID. For more information about the *ImpinjSerializedTID* parameter, see section 4.3.32.

Some tags may not be reported when you use Serialized TID reporting and Monza4-QT tags with both public and short range modes. For more information, see section 5.1 **Speedway Revolution Serialized TID Reporting and Monza4 Tag**.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- None. Extension points are included in the *ImpinjTagReportContentSelector* parameter.

**Definition**

**Table 4.38 ImpinjEnableSerializedTID Parameter**

| ***ImpinjEnableSerializedTID*** Parameter |
|---|
| *SerializedTIDMode*: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Disabled' (default) |
| 1 | 'Enabled' |
| 2-65535 | Reserved for future use |

### 4.3.27 ImpinjEnableRFPhaseAngle Parameter

Use this custom parameter to configure the *ImpinjRFPhaseAngle* feature. When enabled, the *TagReportData* in the RO_ACCESS_REPORT will contain an *ImpinjRFPhaseAngle* parameter that reports the tag's RF Phase Angle. For more information about the *ImpinjRFPhaseAngle* parameter, see section 4.3.33.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- None. Extension points are included in the *ImpinjTagReportContentSelector* parameter.

**Definition**

**Table 4.38 ImpinjEnableRFPhaseAngle Parameter**

| ***ImpinjEnableRFPhaseAngle*** Parameter |
|---|
| *RFPhaseAngleMode*: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Disabled' (default) |
| 1 | 'Enabled' |
| 2-65535 | Reserved for future use |

### 4.3.28 ImpinjEnablePeakRSSI Parameter

Use this custom parameter to configure the *ImpinjPeakRSSI* feature. When enabled, the *TagReportData* in the RO_ACCESS_REPORT will contain an *ImpinjPeakRSSI* parameter that reports the peak RSSI for the tag. For more information about the *ImpinjPeakRSSI* parameter, see section 4.3.34.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- None. Extension points are included in the *ImpinjTagReportContentSelector* parameter.

**Definition**

**Table 4.40 ImpinjEnablePeakRSSI Parameter**

| ***ImpinjEnablePeakRSSI*** Parameter |
|---|
| *PeakRSSIMode*: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Disabled' (default) |
| 1 | 'Enabled' |
| 2-65535 | Reserved for future use |

### 4.3.29 ImpinjEnableGPSCoordinates Parameter

Use this custom parameter to configure the *ImpinjGPSCoordinates* feature. If enabled, and if the GPS receiver has acquired a location fix, the *TagReportData* in the RO_ACCESS_REPORT will contain an *ImpinjGPSCoordinates* parameter. For more information about *ImpinjGPSCoordinates*, see section 4.2.35.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- None. Extensions points are included in the *ImpinjTagReportContentSelector* parameter.

**Definition**

**Table 4.41 ImpinjEnableGPSCoordinates Parameter**

| ***ImpinjEnableGPSCoordinates*** Parameter |
|---|
| *GPSCoordinatesMode*: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Disabled' (default) |
| 1 | 'Enabled' |
| 2-65535 | Reserved for future use |

### 4.3.30 ImpinjEnableOptimizedRead Parameter

Use this custom parameter to configure the *ImpinjOptimizedRead* feature. *ImpinjOptimizedRead* allows the reporting of additional tag memory content during an inventory without the use of AccessSpecs. The Reader optimizes the execution of these reads for improved overall inventory performance.

The memory bank and location of the reads are specified using the *C1G2Read* parameter, just as they are when using AccessSpecs. Similarly, the results of the reads are reported using the *C1G2ReadOpSpecResult* parameter in the *TagReportData* parameter. Reads issued using the *ImpinjOptimizedRead* feature are reported the same as reads using AccessSpecs and thus users should ensure that the OpSpecIDs used for the operations are unique.

Users may configure up to two optimized read operations. One departure from the AccessSpec model is that optimized reads are always attempted, even if the first read fails. So, for example, if the first read results in a failure due to a memory overrun, the second read will still be attempted. Therefore, if there are two optimized reads configured, it is guaranteed that there will be two *C1G2ReadOpSpecResult* parameters in each *TagReportData* parameter generated by the Reader.

Because this feature was designed for optimized inventory performance, any retries configured via the *ImpinjOpSpecRetryCount* parameter do not apply. For more information about the *ImpinjOpSpecRetryCount*, see section 4.3.16.

**LLRP Dependencies**

When the *ImpinjOptimizedRead* feature is enabled, AccessSpecs may still be configured and executed. The results of the AccessSpec execution are reported in the *TagReportData* parameter, after the results of the optimized read. Users should enable the reporting of the AccessSpecID parameter, and use unique *OpSpecIDs* to correlate the results to the actions.

**Allowable Extension Points**

- None. Extension points are included in *ImpinjTagReportContentSelector* parameter.

**Definition**

**Table 4.42 ImpinjEnableOptimizedRead Parameter**

| ***'ImpinjEnableOptimizedRead*** Parameter |
|---|
| *OptimizedReadMode*: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Disabled' (default) |
| 1 | 'Enabled' |
| 2-65535 | Reserved for future use |

**Note:** *C1G2Read*: List of <C1G2Read parameters> [optional, maximum of 2]

### 4.3.31 ImpinjEnableRFDopplerFrequency Parameter

Use this custom parameter to configure the *ImpinjRFDopplerFrequency* feature. If enabled, the *TagReportData* in the RO_ACCESS_REPORT will contain an *ImpinjRFDopplerFrequency* parameter that reports the estimated RF Carrier Doppler shift. For more information about ImpinjRFDopplerFrequency*, see section 4.3.36.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- None. Extension points are included in *ImpinjTagReportContentSelector* parameter.

**Definition**

**Table 4.43 ImpinjEnableRFDopplerFrequency Parameter**

| ***ImpinjEnableRFDopplerFrequency*** Parameter |
|---|
| *RFDopplerFrequencyMode*: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Disabled' (default) |
| 1 | 'Enabled' |
| 2-65535 | Reserved for future use |

### 4.3.32 ImpinjSerializedTID Parameter

Use this custom parameter to report the contents of the tag TID memory bank for Monza 4 tags that support the *ImpinjSerializedTID* feature. Refer to the Monza 4 datasheets to determine which tags support this feature.

Tags inventoried that do not support the feature will omit this parameter from the *TagReportData* in the RO_ACCESS_REPORT. To read the TID memory of tags that don't support this feature, an explicit AccessSpec is required.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- *TagReportData* parameter.

**Definition**

**Table 4.44 ImpinjSerializedTID Parameter**

| ***ImpinjSerializedTID*** Parameter |
|---|
| *TID*: Unsigned Short Array. The contents of the tag TID memory bank.<br>**Custom Extension Point List**: List of <Impinj custom parameter> [optional] |

### 4.3.33 ImpinjRFPhaseAngle Parameter

Use this custom parameter to report the RF phase angle of a singulated tag during normal inventory (EPC backscatter). The *PhaseAngle* field is a scaled, 12-bit value, with **0** representing 0 |deg| (0 radians), and **4096** representing 360 |deg| (2 |pi| radians). For example, if the reported phase angle is 1985, the corresponding angle can be calculated as:

$$1985 \times \frac{360ř}{4096} = 174.46ř \text{ or } 1985 \times \frac{2\pi rad}{4096} = 3.04rad$$

If report accumulation is enabled via the *ROReportSpec* for the currently executing *ROSpec*, the RF phase angle that is reported via this parameter is the phase angle of the last tag singulation. No accumulation of phase data is available.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- TagReportData parameter.

**Definition**

**Table 4.45 ImpinjRFPhaseAngle Parameter**

| ***ImpinjRFPhaseAngle*** Parameter |
|---|
| *PhaseAngle*: Unsigned Short Integer. The scaled phase angle of the tag response during normal inventory. See the Description for a calculation example.<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

### 4.3.34 ImpinjPeakRSSI Parameter

Report the peak RSSI of the tag during the current reporting interval with this custom parameter. Standard LLRP reports peak RSSI in whole dBm units. This parameter provides the same RSSI value in more precise dBm x 100 units. Applications requiring precise RSSI calculations may

enable this parameter instead of (or in addition to) the PeakRSSI LLRP parameter.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- *TagReportData* parameter.

**Definition**

**Table 4.46 ImpinjPeakRSSI Parameter**

| ***ImpinjPeakRSSI*** Parameter |
|---|
| *RSSI*: Signed Short Integer. The peak received power of the EPC backscatter in dBm x 100.<br>**Custom Extension Point List**: List of <*Impinj custom parameter*> [optional] |

### 4.3.35 ImpinjGPSCoordinates Parameter

Use this custom parameter to report the GPS coordinates of the Reader when the tag was singulated . If the GPS receiver has not acquired a location fix, this parameter will not be included in the report. If LLRP accumulation is enabled, the reported coordinates correspond to the last known Reader location when the tag was singulated. The GPS coordinates are reported in signed micro-degrees, so a minor conversion is required to convert the reported value to GPS coordinate formats that are typically used.

For example, if the reported GPS coordinates are 41948240 latitude and -87655562 longitude, this would correspond to (41.948240, -87.655562) or (41? 56' 53.664" N, 87? 39' 20.023" W).

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- *TagReportData* parameter.

**Definition**

**Table 4.47 ImpinjGPSCoordinates Parameter**

| ***ImpinjGPSCoordinates*** Parameter |
|---|
| *Latitude*: Signed Integer. Latitude coordinates in micro-degrees.<br>*Longitude*: Signed Integer. Longitude coordinates in micro-degrees.<br>**Custom Extension Point List**: List of <Impinj custom parameter> [optional] |

### 4.3.36 ImpinjRFDopplerFrequency Parameter

Use this custom parameter to report the estimated RF carrier Doppler frequency shift. The estimate is made over the duration of each tag EPC and has units of Hz. This 16-bit parameter has twelve integer bits and four fractional bits. Accuracy and precision depend on Reader mode and measurement length.

If report accumulation is enabled via the *ROReportSpec* for the currently executing ROSpec, the RF Doppler frequency that is reported by this parameter is the Doppler frequency of the last tag singulation. No accumulation of Doppler frequency data is available.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- *TagReportData* parameter.

**Definition**

**Table 4.48 ImpinjRFDopplerFrequency Parameter**

| ***ImpinjRFDopplerFrequency*** Parameter |
|---|
| *DopplerFrequency*: Signed Short Integer. RF carrier Doppler shift measured over EPC duration.<br>**Custom Extension Point List**: List of <Impinj custom parameter> [optional] |

### 4.3.37 ImpinjLoopSpec Parameter

Use this custom parameter to allow the Reader to loop execution of AISpecs within an ROSpec. If it is included in the list of *SpecParameters* in a ROSpec, it must be the last *SpecParameter* present. There also must be at least one *AISpec* preceding the *ImpinjLoopSpec* parameter. If either condition is not met, the Reader will respond with an error.

**LLRP Dependencies**

This custom parameter overrides the end of a ROSpec. When the last AISpec completes execution, the first AISpec will be executed again until the ROSpec has been executed *LoopCount* iterations.

Allowable Extension Points

- *SpecParameter* parameter.

**Definition**

**Table 4.49 ImpinjLoopSpec Parameter**

| ***ImpinjLoopSpec*** Parameter |
|---|
| *LoopCount*: Unsigned Integer. The number of times to loop execution of the AISpecs of the ROSpec (0 means unlimited).<br>**Custom Extension Point List**: List of <Impinj custom parameter> [optional] |

### 4.3.38 ImpinjGPSNMEASentences Parameter

This custom parameter encapsulates the various NMEA (National Marine Electronic Association) sentences that are supported by the GPS device attached to the Reader. The Sierra Wireless PinPoint XT cellular modem supports GGA and RMC sentences. For more information about NMEA sentences, visit the NMEA 3Twebsite3T.

**xArray**

This custom parameter is not supported by xArray. This parameter will not be reported and any command that uses this parameter will be rejected.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- GET_READER_CONFIG_RESPONSE message.

**Definition**

**Table 4.50 ImpinjGPSNMEASentences Parameter**

| ***ImpinjGPSNMEASentences*** Parameter |
|---|
| *ImpinjGGASentence*: <ImpinjGGASentence Parameter> [optional]<br>*ImpinjRMCSentence*: <ImpinjRMCSentence Parameter> [optional]<br>**Custom Extension Point List**: List of <Impinj custom parameter> [optional] |

### 4.3.39 ImpinjGGASentence Parameter

This custom parameter contains the current GPS information of the Reader's location, as reported in NMEA GGA sentence format. If the GPS device has not acquired a location fix, the string is reported empty.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- None. Extension points are included in *ImpinjGPSNMEASentences* parameter.

**Definition**

**Table 4.51 ImpinjGGASentence Parameter**

| ***ImpinjGGASentence*** Parameter |
|---|
| *GGASentence*: UTF-8 String<br>**Custom Extension Point List**: List of <Impinj custom parameter> [optional] |

### 4.3.40 ImpinjRMCSentence Parameter

This custom parameter contains the current GPS information of the Reader's location as reported in NMEA RMC sentence format. If the GPS device has not acquired a location fix, the string is reported empty.

**LLRP Dependencies**

This custom parameter has no LLRP dependencies.

**Allowable Extension Points**

- None. Extension points are included in ImpinjGPSNMEASentences parameter.

**Definition**

**Table 4.52 ImpinjRMCSentence Parameter**

| ***ImpinjRMCSentence*** Parameter |
|---|
| *RMCSentence*: UTF-8 String<br>**Custom Extension Point List**: List of <Impinj custom parameter> [optional] |

### 4.3.41 ImpinjIntelligentAntennaManagement Parameter

Readers implement the Intelligent Antenna Management feature whereby the Reader looks for the presence of tags on an antenna before proceeding to inventory tags on it. This ensures that no time is wasted on antennas that might not have tags in their field of view. However, in some applications it may be desirable for the Reader to attempt to singulate tags on all selected antennas on a regular basis. This custom parameter allows the user to disable the Intelligent Antenna Management feature.

**LLRP Dependencies**

Intelligent Antenna Management is integral to the Low Duty Cycle feature implemented by Speedway Revolution Readers and xArray Gateways. If Intelligent Antenna Management is disabled, the Low Duty Cycle parameter will be ignored and the feature disabled.

**Allowable Extension Points**

- None. Extension points are included in *ImpinjIntelligentAntennaManagement* parameter.

**Definition**

**Table 4.52 ImpinjIntelligentAntennaManagement Parameter**

| ***ImpinjIntelligentAntennaManagement*** Parameter |
|---|
| *ManagementEnabled*: 1-bit. It set to 1, feature is enabled (default)<br>**Custom Extension Point List**: List of <Impinj custom parameter> [optional] |

### 4.3.42 ImpinjHubConfiguration

This custom parameter displays connected Antenna Hubs. Hubs are identified with IDs 1-4 and in normal operation will display **No_Fault**. If a GPIO Adapter port is not connected to a Hub, the corresponding Hub ID displays **Disconnected**.

**LLRP Dependencies**

If exception events are enabled when an Antenna Hub fault is detected or cleared, an *ImpinjRMCSentence* parameter will be sent in a *ReaderExceptionEvent* parameter.

**Allowable Extension Points**

- GET_READER_CONFIG_RESPONSE message.

**Definition**

**Table 4.53 ImpinjHubConfiguration Parameter**

| ***ImpinjHubConfiguration*** Parameter |
|---|
| *HubID*: Unsigned Short Integer.<br>Connected: Unsigned Short Integer.<br>**Custom Extension Point List**: List of <*Impinj Custom Parameters*> [optional |

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'Unknown' |
| 1 | 'Disconnected' |

| Value | Description |
|---|---|
| 2 | 'Connected' |

Fault: Unsigned Short Integer.

**Possible Values**:

| Value | Description |
|---|---|
| 0 | 'No_Fault' |
| 1 | 'RF_Power' |
| 2 | 'RF_Power_On_Hub_1' |
| 3 | 'RF_Power_On_Hub_2' |
| 4 | 'RF_Power_On_Hub_3' |
| 5 | 'RF_Power_On_Hub_4' |
| 6 | 'No_Init' |
| 7 | 'Serial_Overflow' |
| 8 | 'Disconnected' |

**Note: Custom Extension Point List**: List of <*Impinj custom parameter*> [optional]

### 4.3.43 xArray Parameters

#### ImpinjTiltConfiguration Parameter

This parameter is used to get the Tilt Sensor readings.

**LLRP Dependencies**

This parameter is returned in the GET_READER_CONFIG_RESPONSE if it is specified in an *ImpinjRequestedData* parameter supplied to the GET_READER_CONFIG command. This parameter cannot be used in a SET_READER_CONFIG command.

**Allowable Extension Points**

- GET_READER_CONFIG_RESPONSE message

**Definition**

**Table 4.54 - ImpinjTiltConfiguration Parameter**

| ***ImpinjTiltConfiguration*** Parameter |
|---|
| **XAxis**: Integer - X-Axis orientation in degrees. Possible Values: -90 - +90<br>**YAxis**: Integer - Y-Axis orientation in degrees. Possible Values: -90 - +90<br>**ZAxis**: Integer - Z-Axis orientation in degrees. Possible Values: -90 - +90<br>**Custom Extension Point List**: List of <*Custom Parameters*> [optional] |

### BeaconDurationSeconds Parameter

This parameter is used to get or set the Beacon state of the xArray Gateway. The beacon is a flashing LED on the face of the xArray Gateway that is used to visually identify an xArray Gateway. When the beacon is **ON**, the LED is flashing. When the beacon is **OFF**, the beacon LED is not flashing.

#### LLRP Dependencies

This parameter is returned in the GET_READER_CONFIG_RESPONSE if it is specified in an *ImpinjRequestedData* parameter supplied to the GET_READER_CONFIG command. When used in this manner, the BeaconTimer field will report the configured duration if the beacon has been activated and is currently set to **ON**. When set to **OFF**, the BeaconTime field is zero.

This parameter can be used in a SET_READER_CONFIG command to change the state of the xArray Gateway beacon. When it is used to turn **ON** the beacon, the BeaconDurationSeconds field is used to set the amount of time that the Beacon will be set to "ON" before it is automatically returned to the **OFF** state. Note that a *BeaconDurationSeconds* value of 0 is rejected if *BeaconState* is **ON**.

#### Allowable Extension Points

- GET_READER_CONFIG_RESPONSE message
- SET_READER_CONFIG command

#### Definition

**Table 4.55 - ImpinjBeaconConfiguration Parameter**

| ***ImpinjBeaconConfiguration*** Parameter |
|---|
| **BeaconState**: Boolean - If **TRUE**, then the Beacon is configured to **ON**. If **FALSE**, then the Beacon is configured to **OFF**.<br>**BeaconDurationSeconds**: Unsigned Long Integer - Duration of the beacon in seconds when configured to **ON**.<br>**Custom Extension Point List**: List of <*Custom Parameters*> [optional] |

### ImpinjLocationConfig Parameter

This parameter is used to get or set the configuration of the Location Role of the xArray Gateway device.

#### LLRP Dependencies

This parameter is returned in the GET_READER_CONFIG_RESPONSE if it is specified in an *ImpinjRequestedData* parameter supplied to the GET_READER_CONFIG command. This parameter can be sent in a SET_READER_CONFIG command to configure the xArray Gateway Location Role operation. This parameter can also be sent in a LISpec.

#### Allowable Extension Points

- GET_READER_CONFIG_RESPONSE message
- SET_READER_CONFIG command
- ImpinjLISpec parameter

#### Definition

**Table 4.56 - ImpinjLocationConfig Parameter**

| ***ImpinjLocationConfig*** Parameter |
|---|
| **ComputeWindowSeconds**: Unsigned Short Integer - Duration of the smoothing window in seconds over which tag location estimates are computed.<br>**TagAgeIntervalSeconds**: Unsigned Short Integer - Time in seconds for which the tag must not be read (seen) before it is considered to have exited from the field of view.<br>**UpdateIntervalSeconds**: Unsigned Short Integer - Periodic interval in seconds after which tag location estimates are computed.<br>*Custom Extension Point List*: List of <*Custom Parameters*> [optional] |

### ImpinjC1G2LocationConfig Parameter

This parameter is used to get or set the Gen2 configuration of the Location Role of the xArray Gateway device.

#### LLRP Dependencies

This parameter is returned in the GET_READER_CONFIG_RESPONSE if specified in an *ImpinjRequestedData* parameter supplied to the GET_READER_CONFIG command. This parameter can be sent in a SET_READER_CONFIG command to configure xArray Gateway Location Role operation. This parameter can also be sent in a LISpec.

#### Allowable Extension Points

- GET_READER_CONFIG_RESPONSE message
- SET_READER_CONFIG command
- *ImpinjLISpec* parameter

#### Definition

**Table 4.57 - ImpinjC1G2LocationConfig Parameter**

| ***ImpinjC1G2LocationConfig*** Parameter |
|---|
| **ModeIndex**: Unsigned Short Integer - The Gen2 mode used by the Location Role<br>**Session**: Unsigned 2-bit Integer - The Gen2 session used by the Location Role<br>**C1G2Filter**: List of <*C1G2Filter Parameters*>[optional] to filter tags addressed by the Location Role<br>**Custom Extension Point List**: List of <*Custom Parameters*> [optional] |

### ImpinjLocationReporting Parameter

This parameter is used to get or set the configuration of reporting of events in during Location Role operation.

#### LLRP Dependencies

This parameter is returned in the GET_READER_CONFIG_RESPONSE if specified in an *ImpinjRequestedData* parameter supplied to the GET_READER_CONFIG command. This parameter can be sent in a SET_READER_CONFIG command to configure xArray Location Role reporting. This parameter can also be sent in a LISpec.

#### Allowable Extension Points

- GET_READER_CONFIG_RESPONSE message
- SET_READER_CONFIG command
- *ImpinjLISpec* parameter

#### Definition

**Table 4.58 - ImpinjLocationReporting Parameter**

| ***ImpinjLocationReporting*** Parameter |
|---|
| **EnableUpdateReport**: Boolean - When true, reports are generated immediately after the tag location estimates are computed at the interval specified by the *UpdateIntervalSeconds* field of the *ImpinjLocationConfig* parameter. |

| ***ImpinjLocationReporting*** Parameter |
|---|
| **EnableEntryReport**: Boolean - When true, a report is generated when a tag enters the field of view (is seen for the first time).<br>**EnableExitReport**: Boolean - When true, a report is generated when a tag exits the field of view, as qualified by the *TagAgeIntervalSeconds* field of the *ImpinjLocationConfig* parameter.<br>**EnableDiagnosticReport**: Boolean - When true, diagnostic reports are generated.<br>**Custom Extension Point List**: List of <Custom Parameters> [optional] |

### ImpinjPlacementConfiguration Parameter

This parameter is used to get or set the mounting location of the xArray Gateway.

#### LLRP Dependencies

This parameter is returned in the GET_READER_CONFIG_RESPONSE if it is specified in an *ImpinjRequestedData* parameter supplied to the GET_READER_CONFIG command. This parameter can be used in a SET_READER_CONFIG command to configure the xArray mounting location.

#### Allowable Extension Points

- GET_READER_CONFIG_RESPONSE message
- SET_READER_CONFIG command

#### Definition

**Table 4.59 - ImpinjPlacementConfiguration Parameter**

| ***ImpinjPlacementConfiguration*** Parameter |
|---|
| **HeightCm**: Unsigned Short Integer - Height in centimeters with respect to the average tag height.<br>**FacilityXLocationCm**: Integer - The relative position of the xArray in centimeters within the facility. This is used by the xArray when computing location and might be useful for multi-xArray deployments.<br>**FacilityYLocationCm**: Integer - The relative position of the xArray in centimeters within the facility. This is used by the xArray when computing location and might be useful for multi-xArray deployments.<br>**OrientationDegrees**: Short Integer - The relative orientation of the xArray X-Y coordinates relative to the Store X-Y coordinates in degrees.<br>**Custom Extension Point List**: List of <*Custom Parameters*> [optional] |

## 4.4 Operation

### 4.4.1 xArray

#### Standard Parameters

##### GPITriggerValue Parameter

GPI triggering is not supported by xArray. This parameter will not be reported and any command that uses this parameter will be rejected.

##### AISpecStopTrigger Parameter

*AIspecStopTriggerType* cannot be **2**. GPI triggering is not supported by xArray.

##### ROSpecStopTrigger Parameter

*ROSpecStopTriggerType* cannot be **2**. GPI triggering is not supported by xArray.

##### EventNotificationState Parameter

The following values of *EventType* will not be reported and any command that uses these values will be ignored:

- 1 - GPIEvent

##### GPIEvent Parameter

This parameter is not supported by xArray. This parameter will not be reported and any command that uses this parameter will be rejected.

#### xArray Custom Parameters

##### ImpinjLISpec Parameter

This parameter is used to configure Location Role operation of the xArray Gateway. Only one *ImpinjLISpec* parameter can be specified. If an *ImpinjLISpec* parameter is specified, no other *AISpec*, or *RFSurveySpec* may be additionally specified.

###### LLRP Dependencies

When the xArray Gateway is set to operate in the Location Role, all settings for the operation are conveyed by:

- *ImpinjLocationConfig*
- *ImpinjC1G2LocationConfig*
- *ImpinjLocationReporting*

If no values for these parameters are provided in the *ImpinjLISpec* and no values of these parameters are provided in SET_READER_CONFIG, default values are used. Configuration from the following parameters does not apply during location inventory:

- *C1G2RFControl* parameter
- *RFTransmitter* parameter
- *C1G2Filter* parameter
- *C1G2SingulationControl* parameter
- *C1G2InventoryCommand* parameter

**Allowable Extension Points**

- *ROSpec* Parameter

**Definition**

**Table 4.60 - ImpinjLISpec Parameter**

| ***ImpinjLISpec*** Parameter |
|---|
| **ImpinjLocationConfig**: <*ImpinjLocationConfig Parameter*> [optional]<br>**ImpinjC1G2LocationConfig**: <*ImpinjC1G2LocationConfig* Parameter> [optional]<br>**ImpinjLocationReporting**: <*ImpinjLocationReporting* Parameter> [optional]<br>**Custom Extension Point List**: List of <*Custom Parameters*> [optional] |

### xArray Reports and Notifications

The following two parameters are ignored by the xArray during Location Role operations:

- *ROReportSpec* parameter
- *ImpinjTagReportContentSelector*

### ImpinjExtendedTagInformation Parameter

This custom parameter provides additional information about one or more tags that cannot be conveyed by using the standard LLRP *TagReportData* parameter. Following the EPCs is a set of parameters that provide additional information about the tag, such as its location or its direction of travel.

**LLRP Dependencies**

None

**Allowable Extension Points**

- RO_ACCESS_REPORT

**Definition**

**Table 4.61 - ImpinjExtendedTagInformation Parameter**

| ***ImpinjExtendedTagInformation*** Parameter |
|---|
| **EPCData**: List of <*EPCData* Parameter> - one or more EPCs about which the report applies<br>**ImpinjLocationReportData**: <*ImpinjLocationReportData* Parameter> [optional]<br>**Custom Extension Point List**: List of <*Custom Parameters*> [optional] |

### ImpinjLocationReportData Parameter

This custom parameter provides tag location information. It may be present in the report when the xArray is running the Location Role and the report is enabled.

**LLRP Dependencies**

None

**Allowable Extension Points**

- *ImpinjExtendedTagInformation*

**Definition**

**Table 4.62 - ImpinjLocationReportData Parameter**

| ***ImpinjLocationReportData*** Parameter |
|---|
| **LastSeenTimestampUTC**: Unsigned 64-bit integer - the UTC timestamp at which the tag was last seen (read).<br>**LocXCentimeters**: Integer - the location of the tag on the facility X axis.<br>**LocyCentimeters**: Integer - the location of the tag on the facility Y axis.<br>**Type**: enumeration - the report type (0 - Entry, 1 - Update, 2 - Exit)<br>**ImpinjLocationConfidence**: <*ImpinjLocationConfidence* Parameter><br>**Custom Extension Point List**: List of <*Custom Parameters*> [optional] |

### ImpinjLocationConfidence Parameter

This custom parameter defines the confidence information for a location report.

**LLRP Dependencies**

None

**Allowable Extension Points**

- *ImpinjLocationReportData*

**Definition**

**Table 4.63 - ImpinjLocationConfidence Parameter**

| ***ImpinjLocationConfidence*** Parameter |
|---|
| **ReadCount**: Unsigned Short Integer - the number of read observations used for this location estimate<br>**ConfidenceData**: Array of Unsigned Integers - Diagnostic Confidence data array<br>**Custom Extension Point List**: List of <*Custom Parameters*> [optional] |

# 5 Advanced Topics

## 5.1 Speedway Revolution Serialized TID Reporting and Monza4 Tags

Some tags might not be reported when you use Serialized TID reporting and Monza4-QT tags with both public and short range modes. Speedway Revolution has an optional Serialized TID reporting feature that is intended to provide more information efficiently. The Monza4 has an optional public, short range feature that is intended to provide less information for privacy reasons. When the two features are used together, some tags might not be reported, yet everything is working exactly as intended.

The Speedway Revolution Serialized TID reporting feature – EPC+TID – causes the Reader to ask the tags for their Serialized TID during routine inventory operations. Usually tags are asked only for their EPC. By asking the tags to immediately send their TID, fewer interactions with the tag are needed to obtain both EPC and TID, and performance or the number of tags per second is better. When EPC+TID reporting is enabled, the Reader strives to consistently report both EPC and TID.

If a tag responds with only the EPC, the Reader immediately issues a read TID operation. If a transient error occurs the tag is skipped, not reported, and is retried later. If a persistent error occurs, the Reader reports only the EPC to the application.

The Monza4 tag has two independent modes, which are both intended to protect privacy. While a Monza4 tag is in **public** mode, it intentionally does not provide the TID when asked for EPC+TID. While a Monza4 tag is in **short range** mode, it intentionally does not support certain operations, including a read TID operation, unless it is close to the Reader antenna.

Specifically, The Reader requests EPC+TID from a Monza4 tag in **public** mode and **short range** mode. In this case, the Monza4 tag will only respond with the EPC. The Reader will immediately try to read the TID. If the Monza4 is close to the antenna, the TID read operation will work, which means that the EPC and TID are reported to the application. However, if the Monza4 tag is over a meter from the antenna, the TID read operation is simply ignored. The Monza4 tag deems reading the tag as a possible privacy invasion, and the Reader interprets the lack of tag response as a transient error **not reporting the tag**, unless a retry is successful. The retry can't be successful until the tag is brought close to the antenna read zone.

# 6 Custom Extension Encoding

This section describes the encoding of Impinj custom extensions.

## 6.1 Custom Messages

All LLRP custom messages are encoded with a common header to ensure a unique namespace across all LLRP implementations. This header appears in each of the custom messages below for completeness. Byte and bit order are shown as the first and second row of the table respectively. The vendor ID field contains the Impinj Private Enterprise Number (PEN) - 25882. A unique subtype indicator defines each custom extension message.

**Table 6-1 Impinj Custom Message Header**

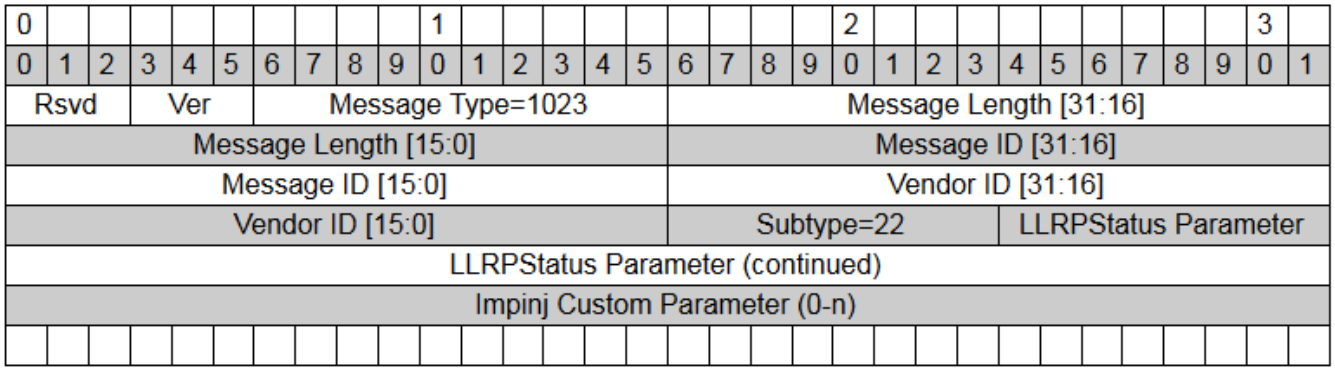
| 0 | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| Rsvd | | | Ver | | | Message Type=1023 | | | | | | | | | | Message Length [31:16] | | | | | | | | | | | | | | | |
| Message Length [15:0] | | | | | | | | | | | | | | | | Message ID [31:16] | | | | | | | | | | | | | | | |
| Message ID [15:0] | | | | | | | | | | | | | | | | Vendor ID [31:16]= **25882** | | | | | | | | | | | | | | | |
| Vendor ID (continued) | | | | | | | | | | | | | | | | Subtype=Varies | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

### 6.1.1 IMPINJ_ENABLE_EXTENSIONS

| 0 | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| Rsvd | | | Ver | | | Message Type=1023 | | | | | | | | | | Message Length [31:16] | | | | | | | | | | | | | | | |
| Message Length [15:0] | | | | | | | | | | | | | | | | Message ID [31:16] | | | | | | | | | | | | | | | |
| Message ID [15:0] | | | | | | | | | | | | | | | | Vendor ID [31:16] | | | | | | | | | | | | | | | |
| Vendor ID [15:0] | | | | | | | | | | | | | | | | Subtype=21 | | | | | | | | Reserved [31:24] | | | | | | | |
| Reserved [23:0] | | | | | | | | | | | | | | | | | | | | | | | | Impinj Cust. Parameter | | | | | | | |
| Impinj Custom Parameter (0-n) (continued) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

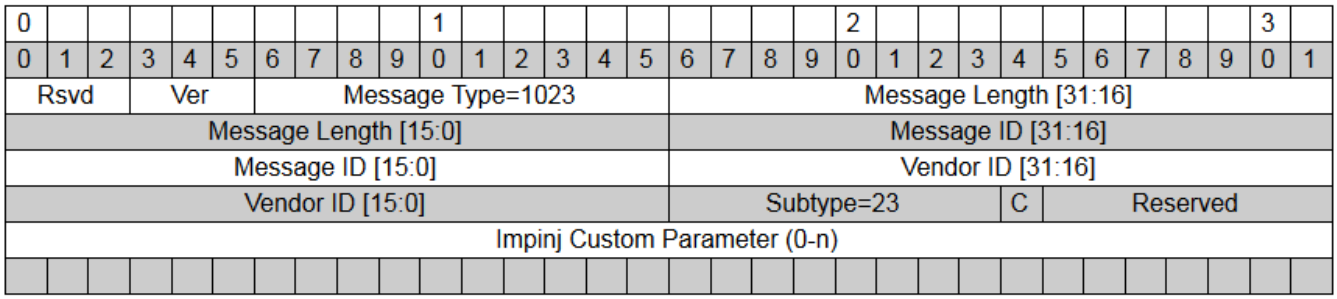
For more information, see section 4.2.1.

### 6.1.2 IMPINJ_ENABLE_EXTENSIONS_RESPONSE




For more information, see section 4.2.2.

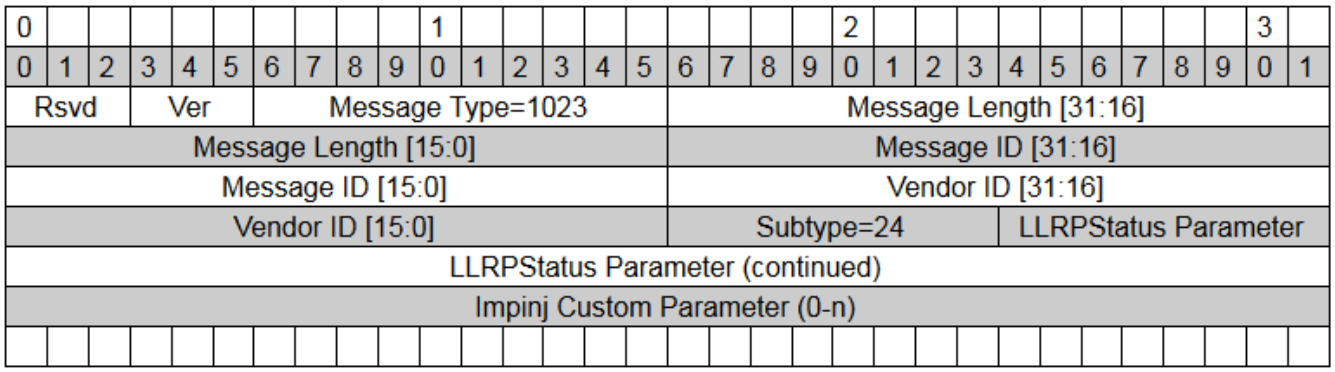
### 6.1.3 IMPINJ_SAVE_SETTINGS




Abbreviations: C Save Configuration

For more information, see section 4.2.3.

### 6.1.4 IMPINJ_SAVE_SETTINGS_RESPONSE



For more information, see section 4.2.4.

## 6.2 Custom Parameters

All LLRP custom parameters are encoded with a common header to ensure a unique namespace across all LLRP implementations. This header appears in each of the custom parameters below for completeness. Byte and bit order are shown as the first and second row of the table respectively. The vendor ID field contains the Impinj Private Enterprise Number (PEN) - 25882. A unique subtype indicator defines each custom extension parameter.

| 0 | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| Reserved | | | | | | Type=1023 | | | | | | | | | | Parameter Length | | | | | | | | | | | | | | | |
| Vendor ID=25882 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Subtype=Varies | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

### 6.2.1 ImpinjRequestedData Parameter

| 0 | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| Reserved | | | | | | Type=1023 | | | | | | | | | | Parameter Length | | | | | | | | | | | | | | | |
| Vendor ID=25882 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Subtype=21 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| RequestedData | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Impinj Custom Parameter (0-n) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

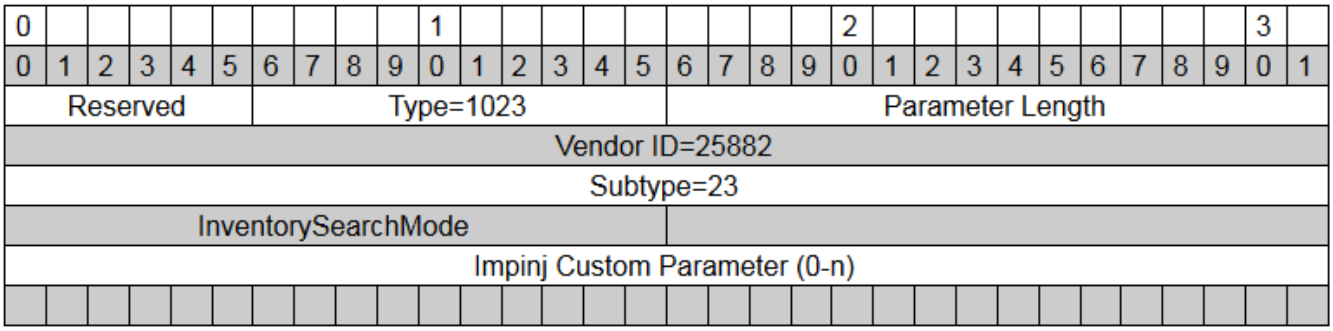
For more information, see section 4.3.1.

### 6.2.2 ImpinjSubRegulatoryRegion Parameter

| 0 | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| Reserved | | | | | | Type=1023 | | | | | | | | | | Parameter Length | | | | | | | | | | | | | | | |
| Vendor ID=25882 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Subtype=22 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| RegulatoryRegion | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Impinj Custom Parameter (0-n) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

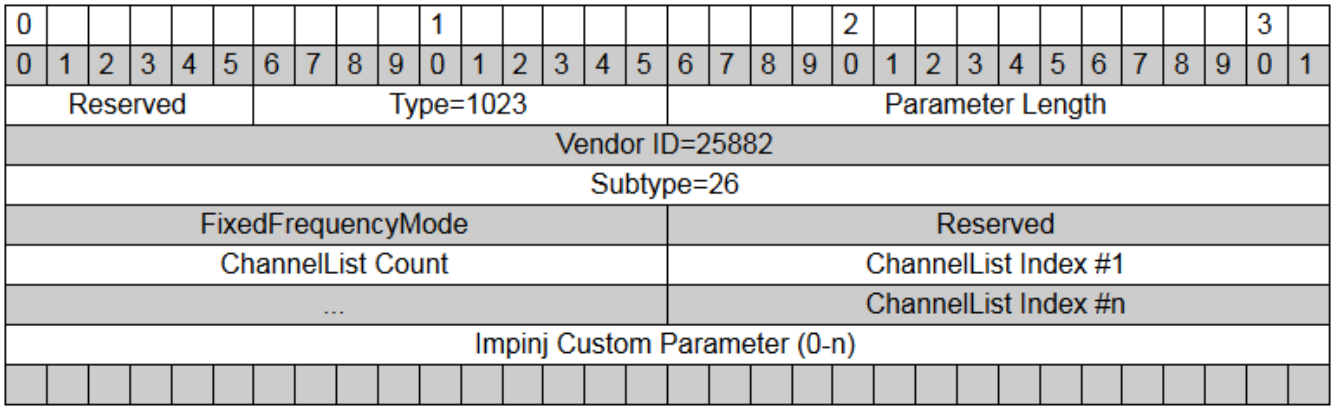
For more information, see section 4.3.2.

### 6.2.3 ImpinjInventorySearchMode Parameter



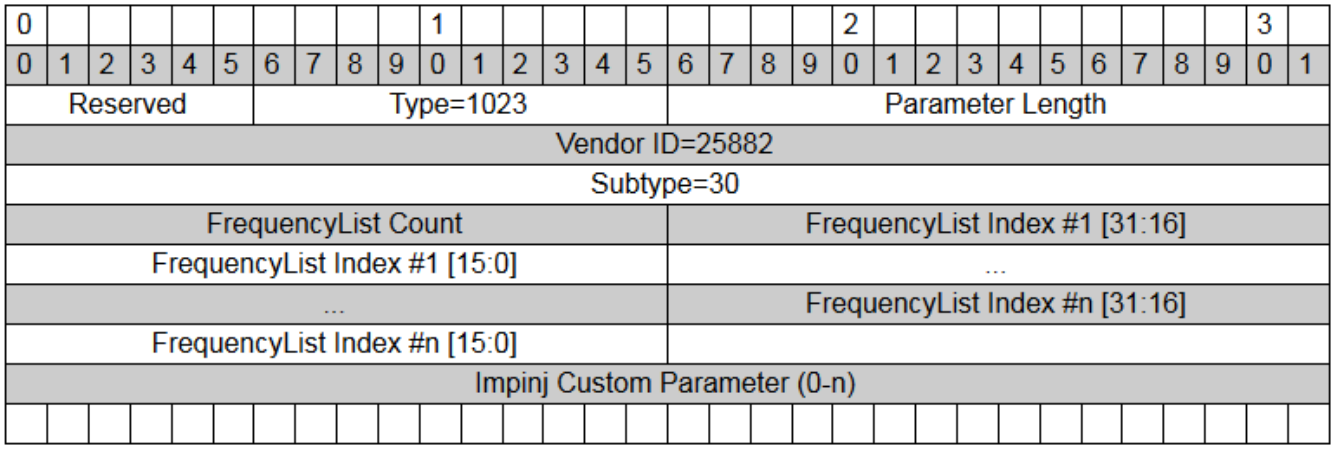

For more information, see section 4.3.3.

### 6.2.4 ImpinjFixedFrequencyList Parameter



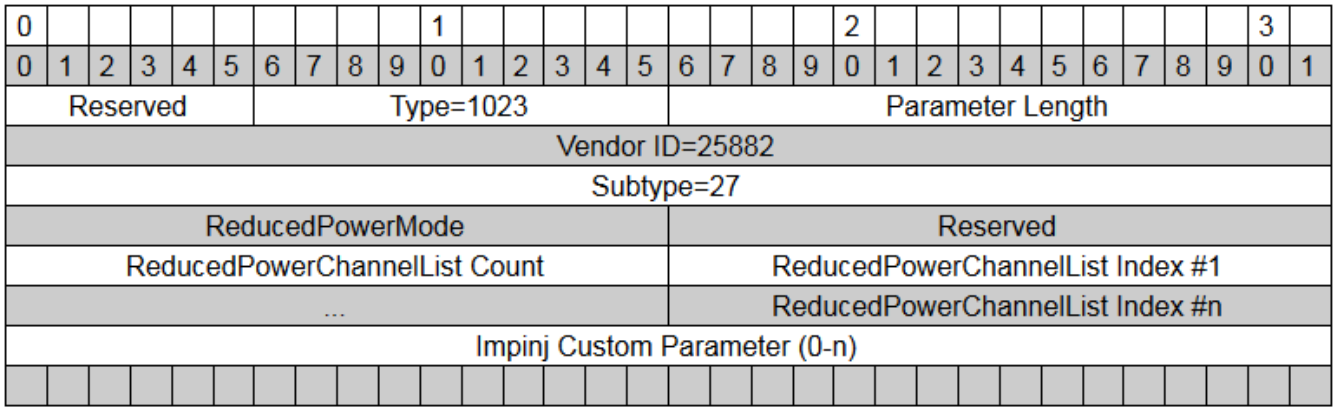

For more information, see section 4.3.4.

### 6.2.5 ImpinjFrequencyCapabilities Parameter



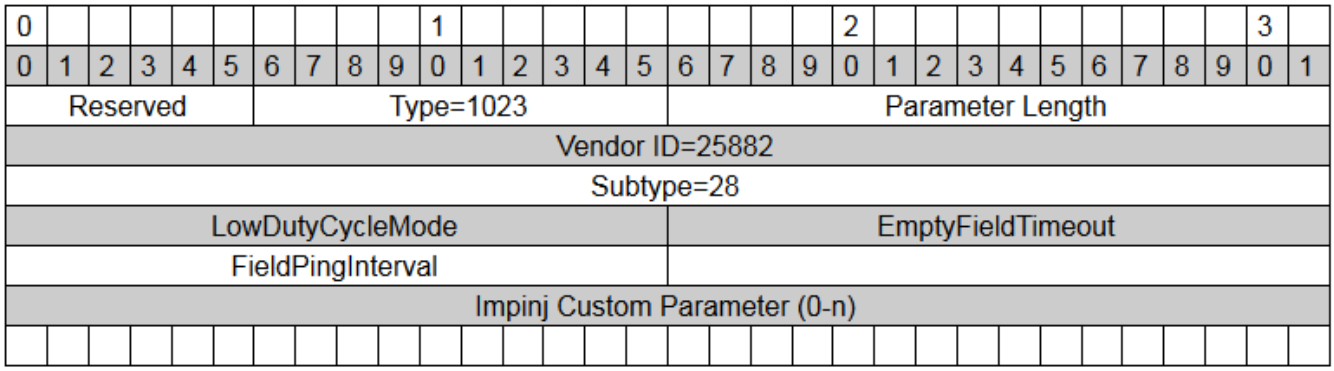

For more information, see section 4.3.5

### 6.2.6 ImpinjReducedPowerFrequencyList Parameter



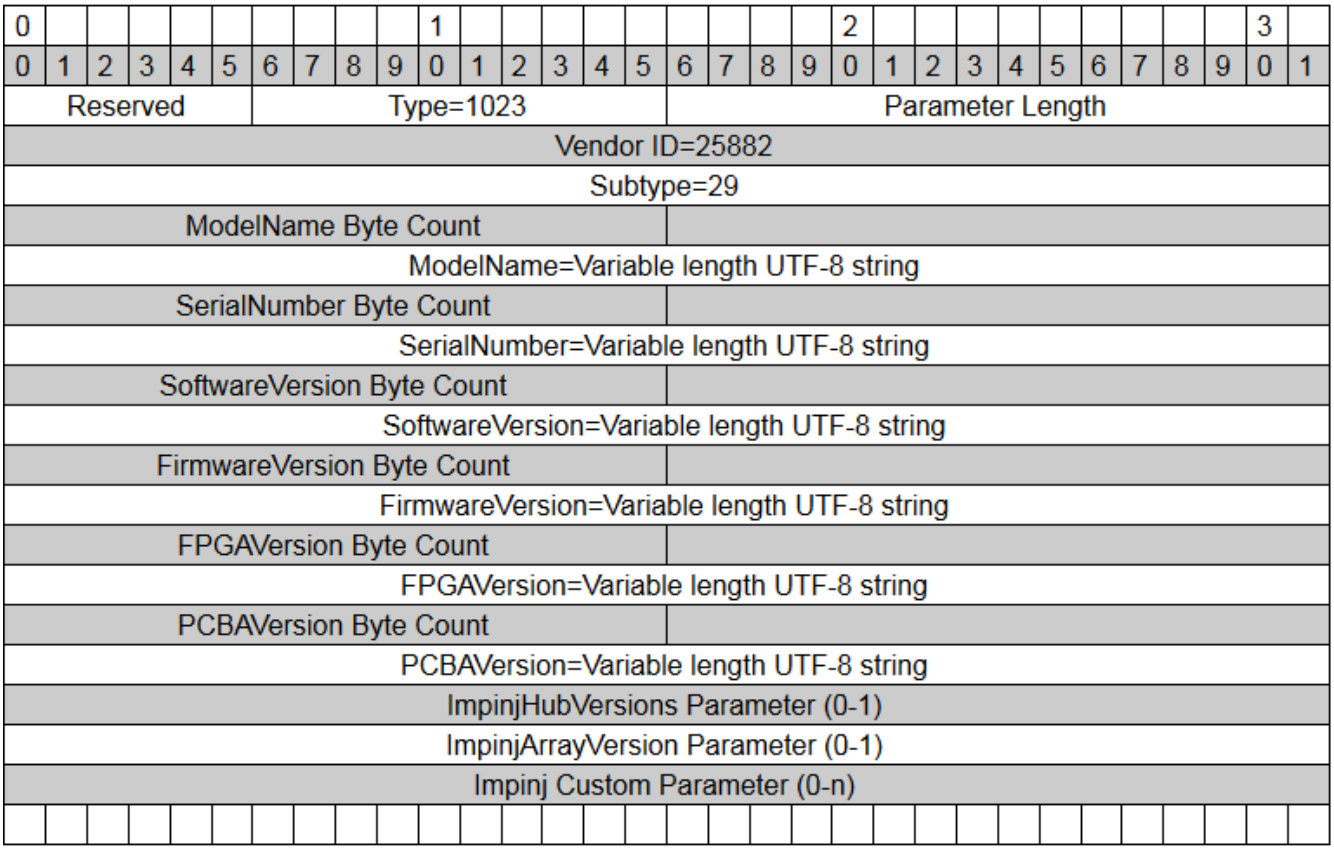

For more information, see section 4.36.

### 6.2.7 ImpinjLowDutyCycle Parameter



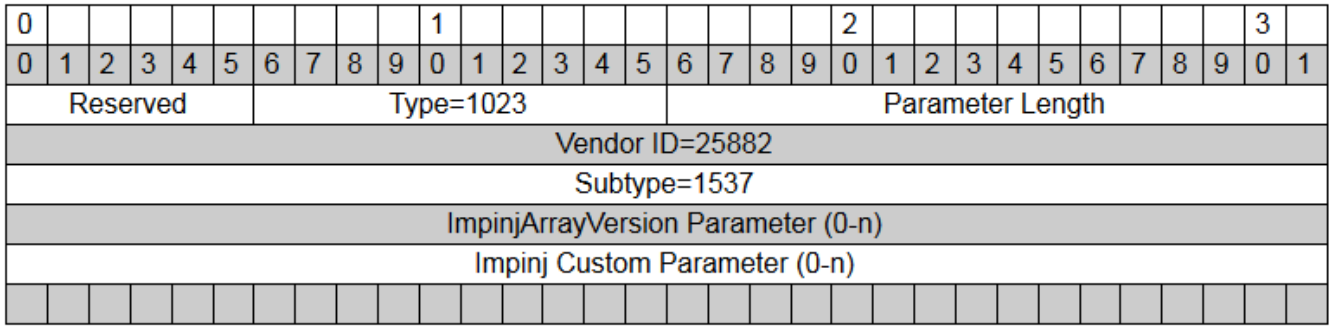

For more information, see section 4.3.7.

### 6.2.8 ImpinjDetailedVersion Parameter



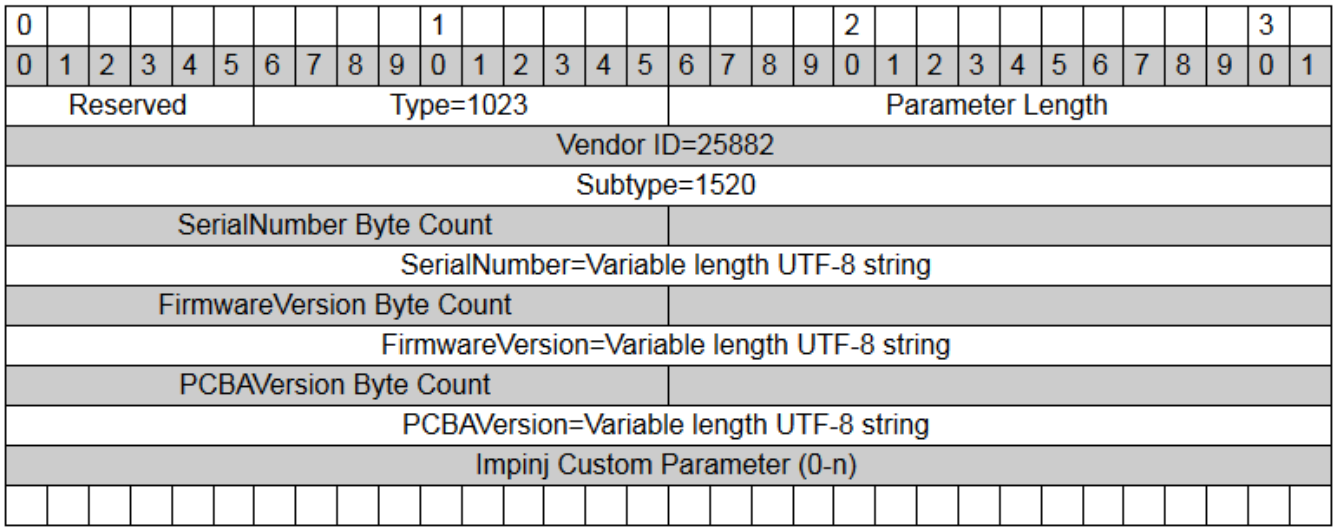

For more information, see section 4.3.8.

### 6.2.9 ImpinjHubVersions Parameter



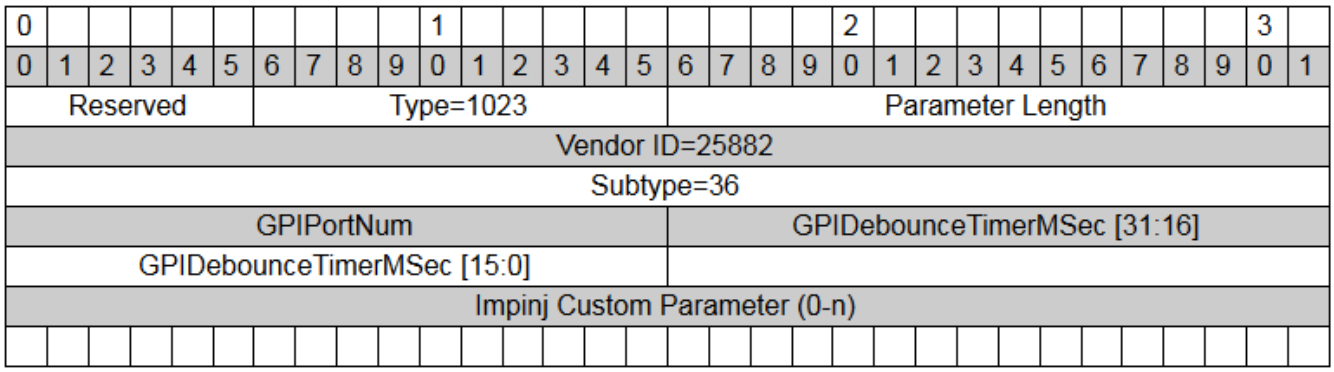

For more information, see section 4.3.8.

### 6.2.10 ImpinjArrayVersion Parameter



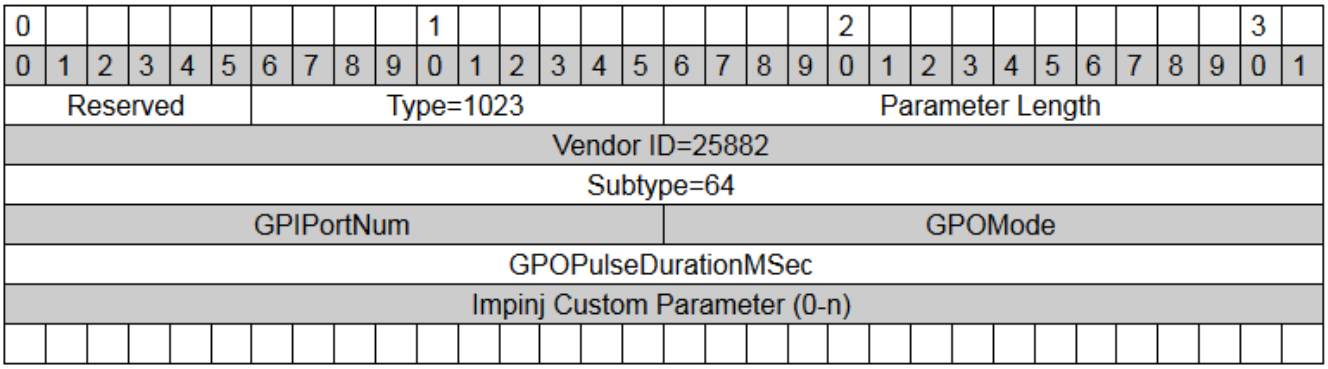

For more information, see section 4.3.8.

### 6.2.11 ImpinjGPIDebounceConfiguration Parameter



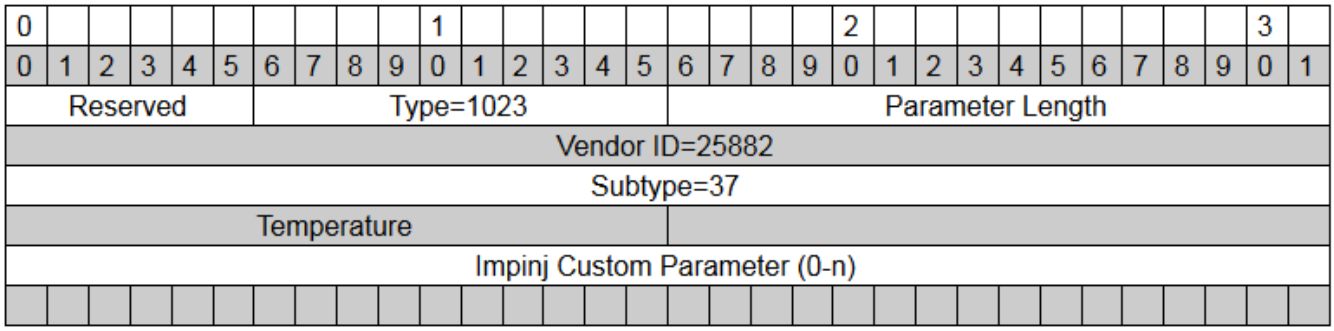

For more information, see section 4.3.9.

### 6.2.12 ImpinjAdvancedGPOConfiguration Parameter



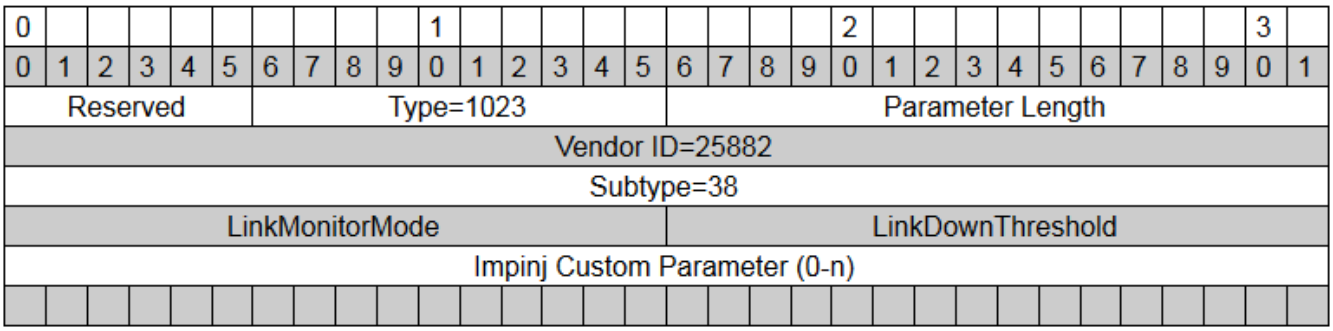

For more information, see section 4.3.10.

### 6.2.13 ImpinjReaderTemperature Parameter



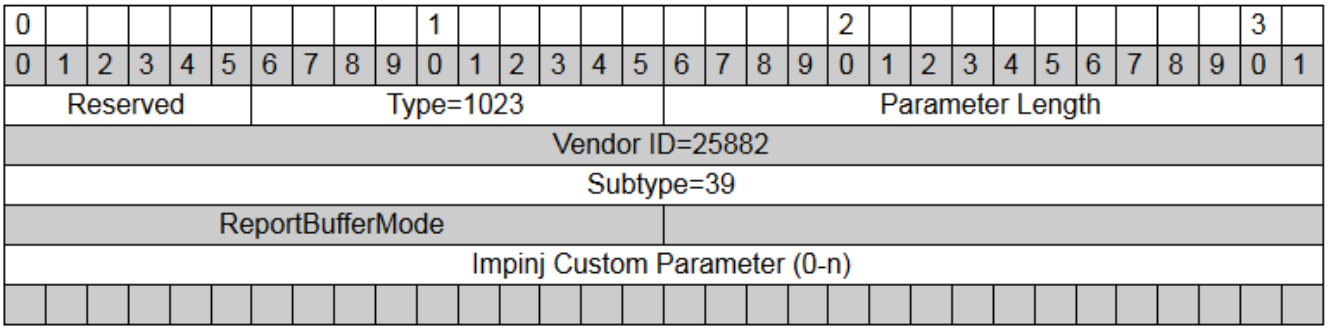

For more information, see section 4.3.11.

### 6.2.14 ImpinjLinkMonitorConfiguration Parameter



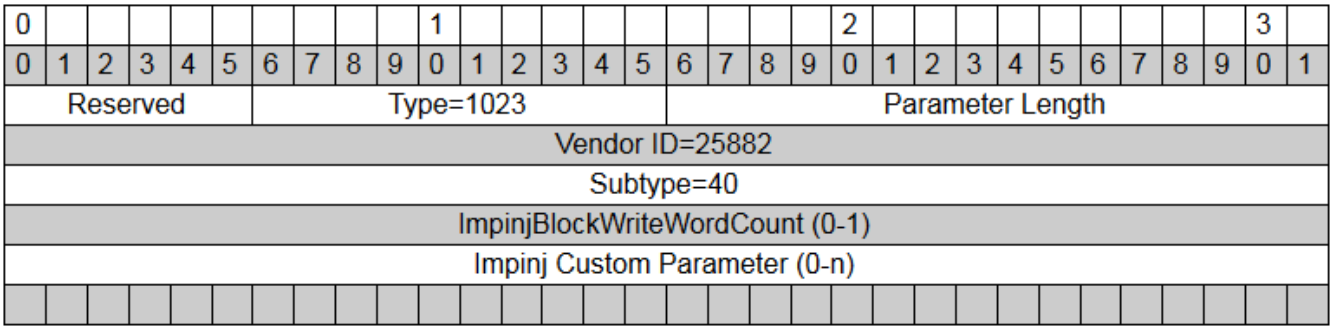

For more information, see section 4.3.12.

### 6.2.15 ImpinjReportBufferConfiguration Parameter



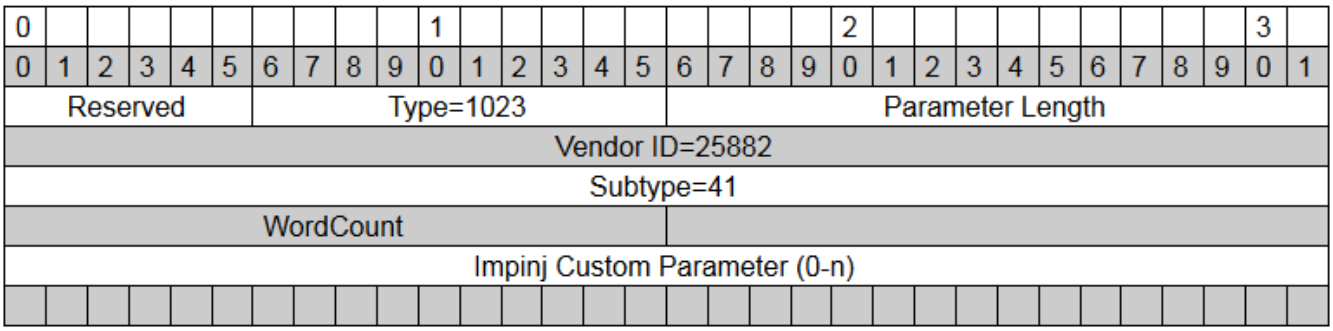

For more information, see section 4.3.13.

### 6.2.16 ImpinjAccessSpecConfiguration Parameter



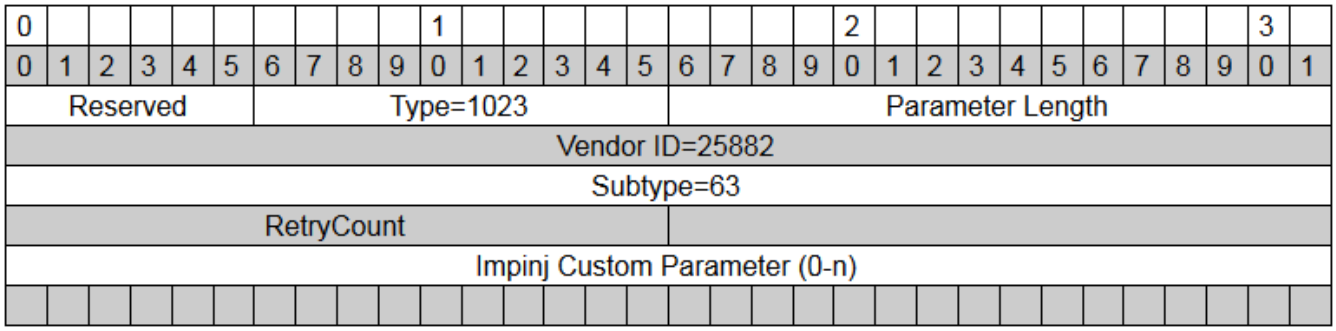

For more information, see section 4.3.14.

### 6.2.17 ImpinjBlockWriteWordCount Parameter



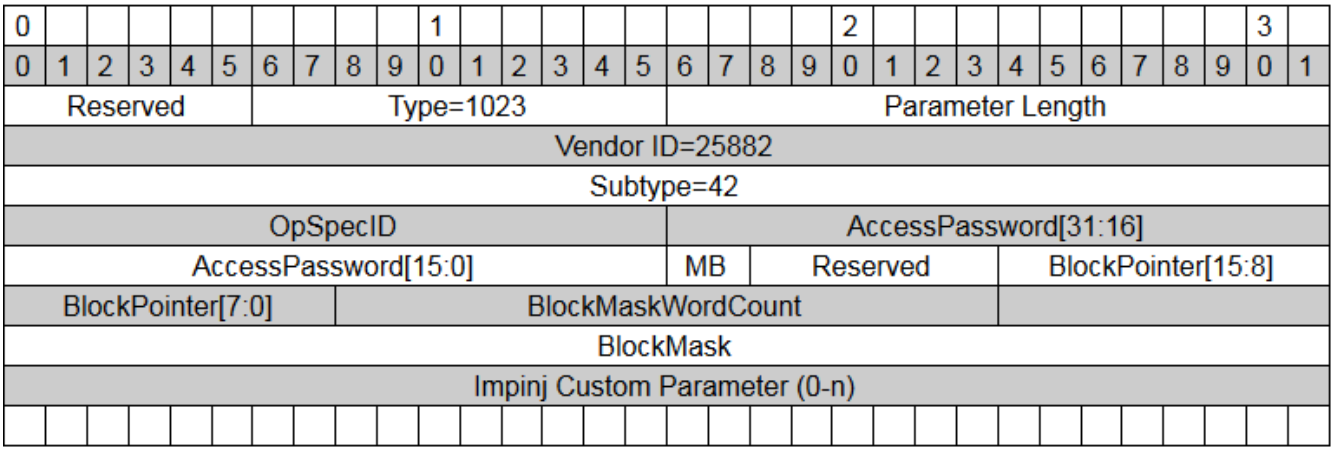

For more information, see section 4.3.15.

### 6.2.18 ImpinjOpSpecRetryCount Parameter



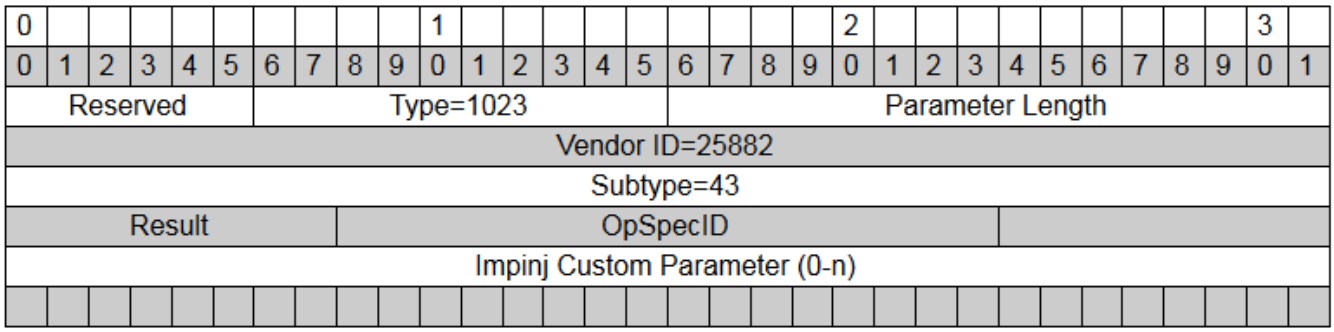

For more information, see section 4.3.16.

### 6.2.19 ImpinjBlockPermalock Parameter



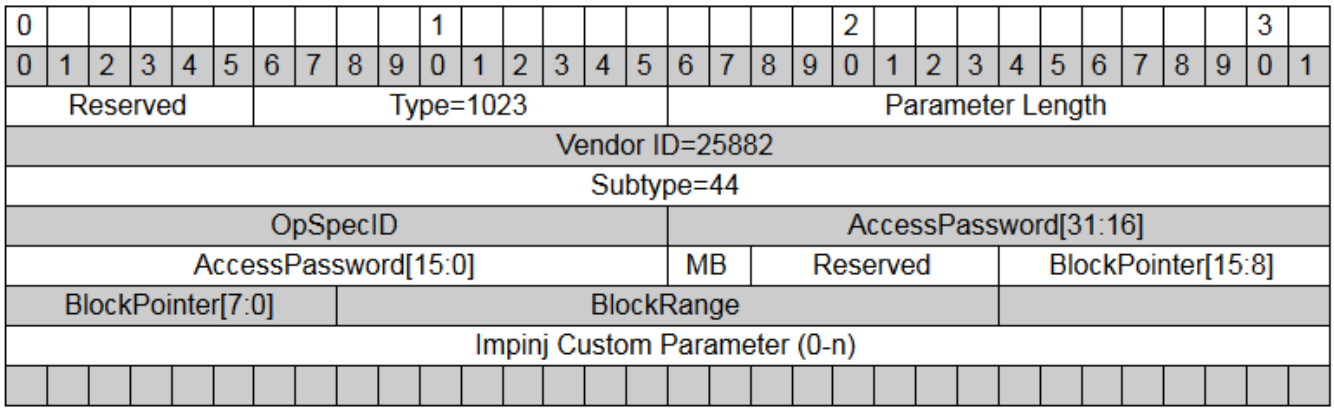
For more information, see section 4.3.17.

### 6.2.20 ImpinjBlockPermalockOpSpecResult Parameter



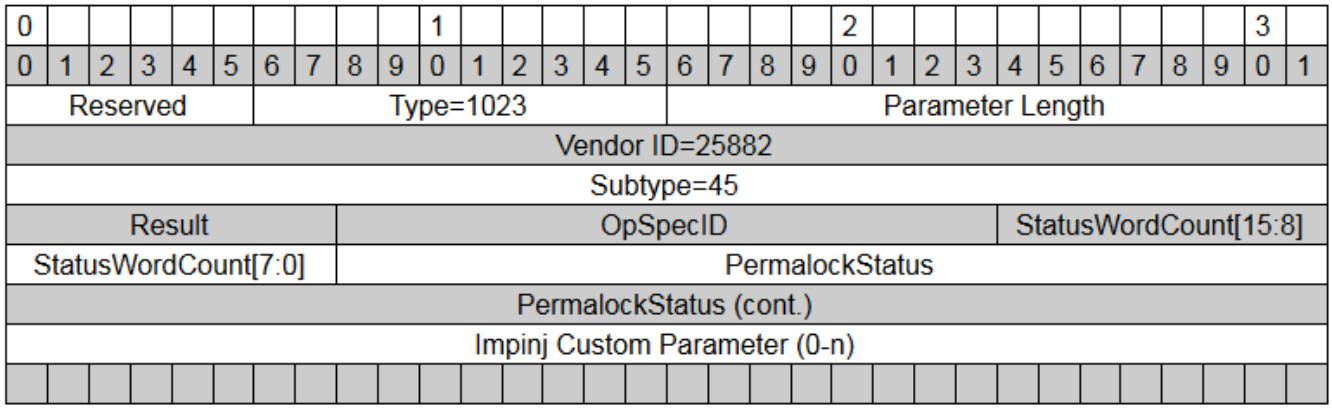

For more information, see section 4.3.18.

### 6.2.21 ImpinjGetBlockPermalockStatus Parameter



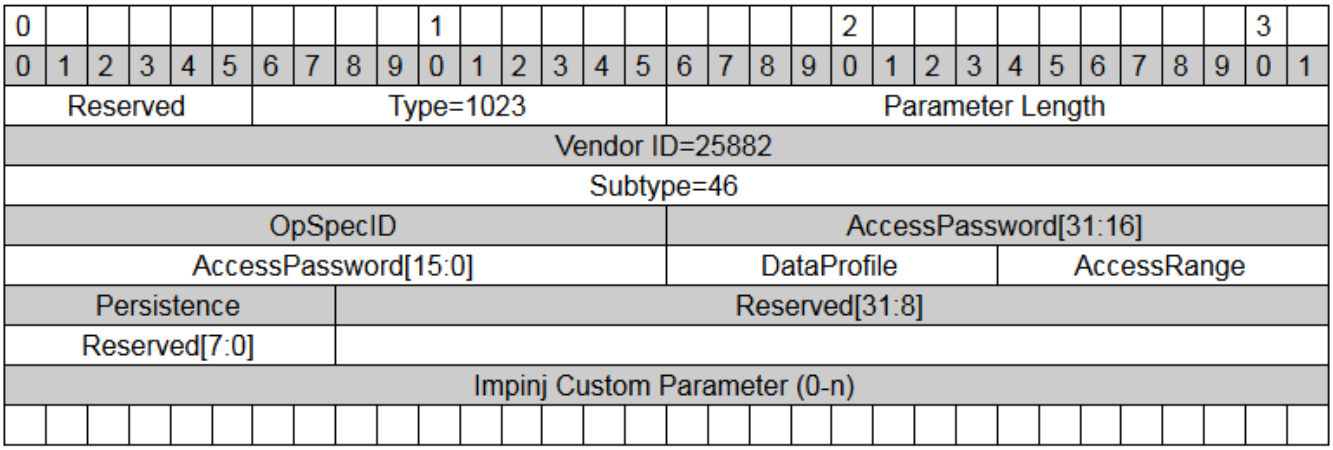

For more information, see section 4.3.19.

### 6.2.22 ImpinjGetBlockPermalockStatusOpSpecResult Parameter



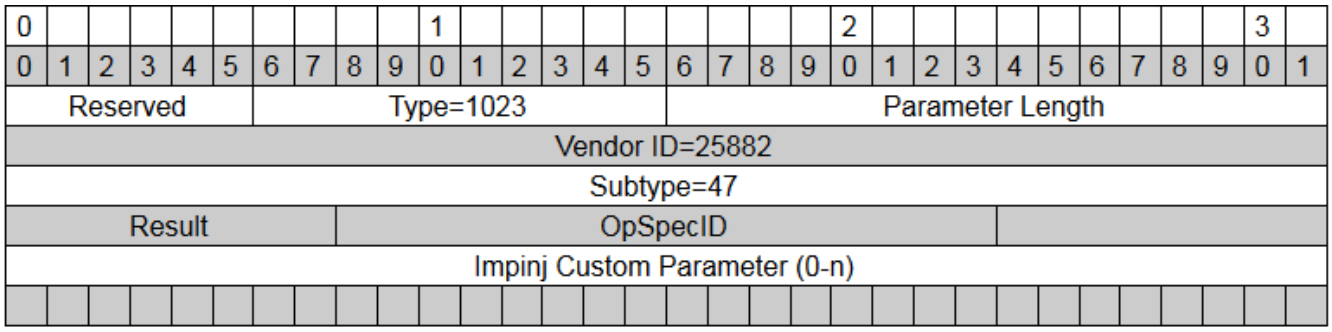

For more information, see section 4.3.20.

### 6.2.23 ImpinjSetQTConfig Parameter



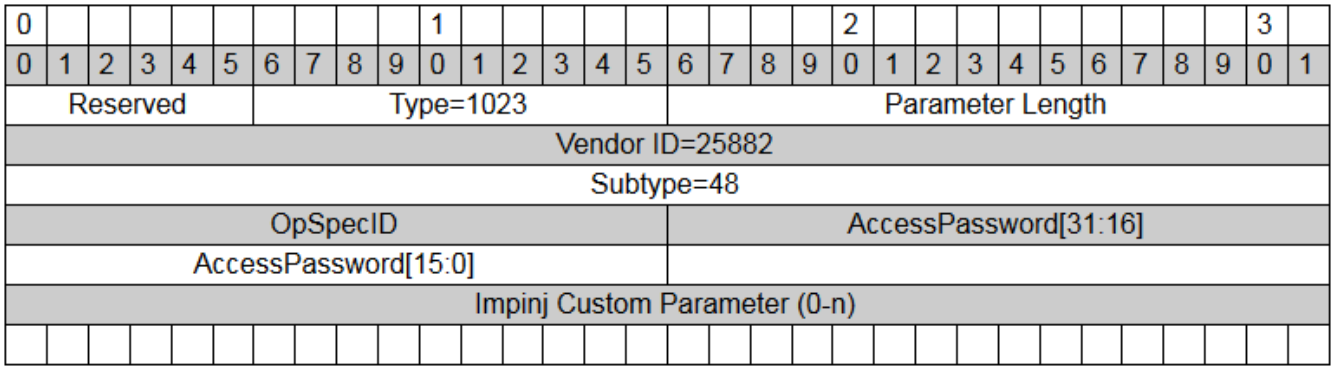

For more information, see section 4.3.21.

### 6.2.24 ImpinjSetQTConfigOpSpecResult Parameter



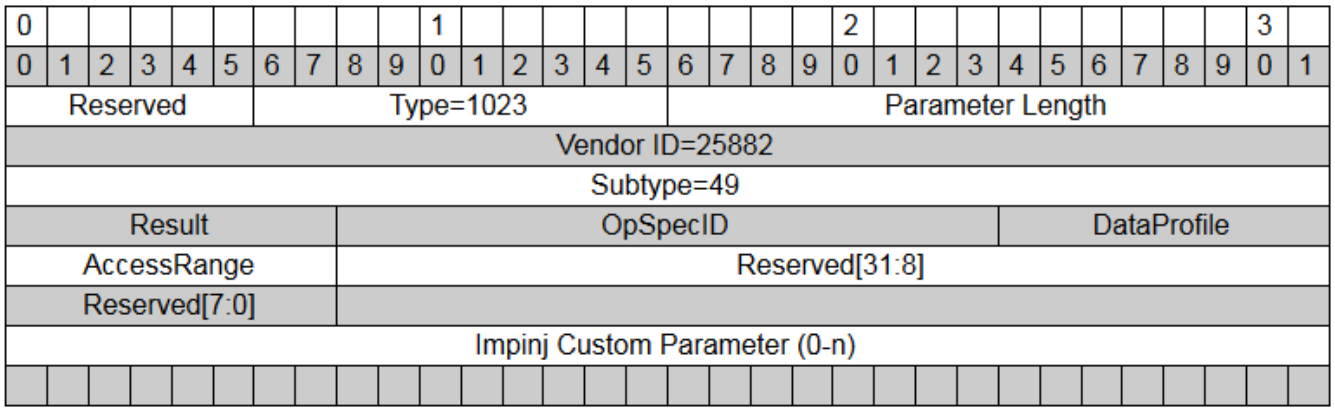

For more information, see section 4.3.22.

### 6.2.25 ImpinjGetQTConfig Parameter



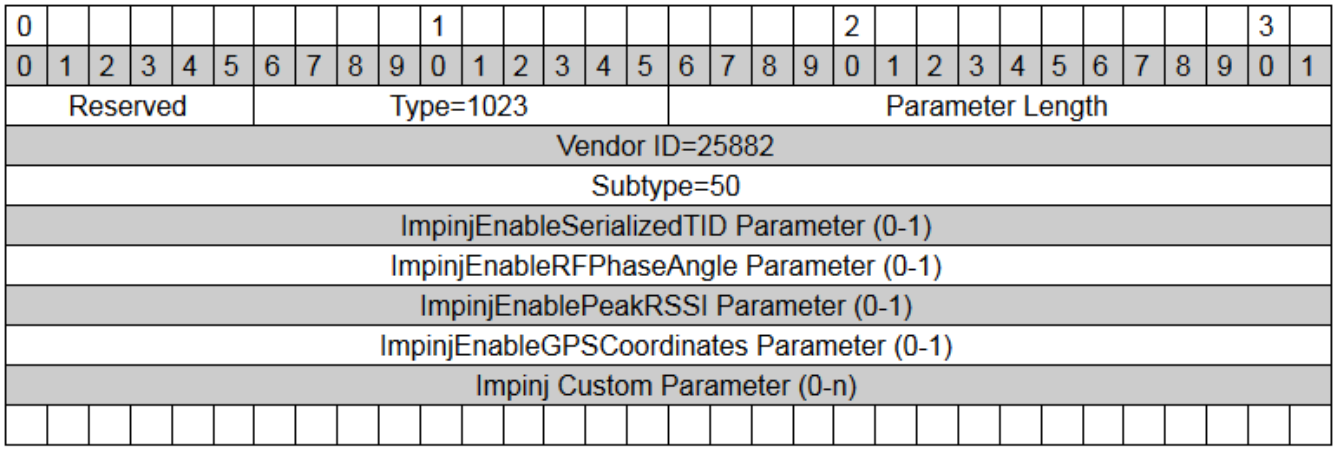

For more information, see section 4.3.23.

### 6.2.26 ImpinjGetQTConfigOpSpecResult Parameter



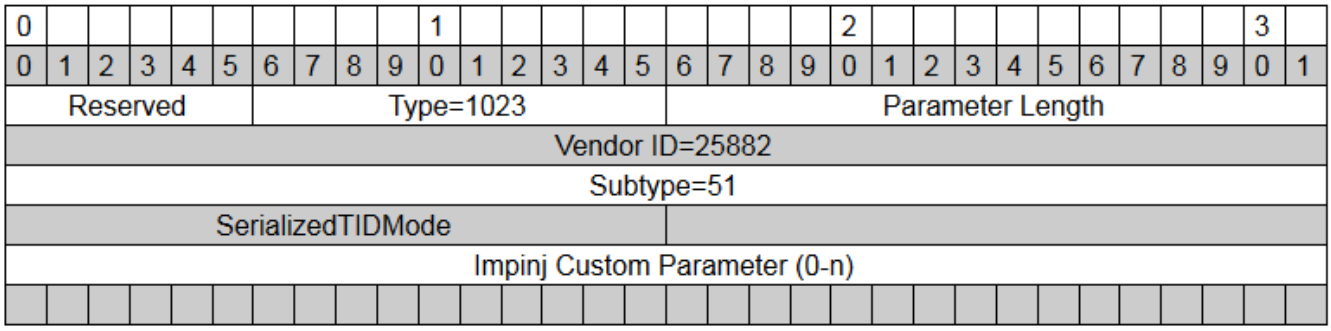
For more information, see section 4.3.24.

### 6.2.27 ImpinjTagReportContentSelector Parameter



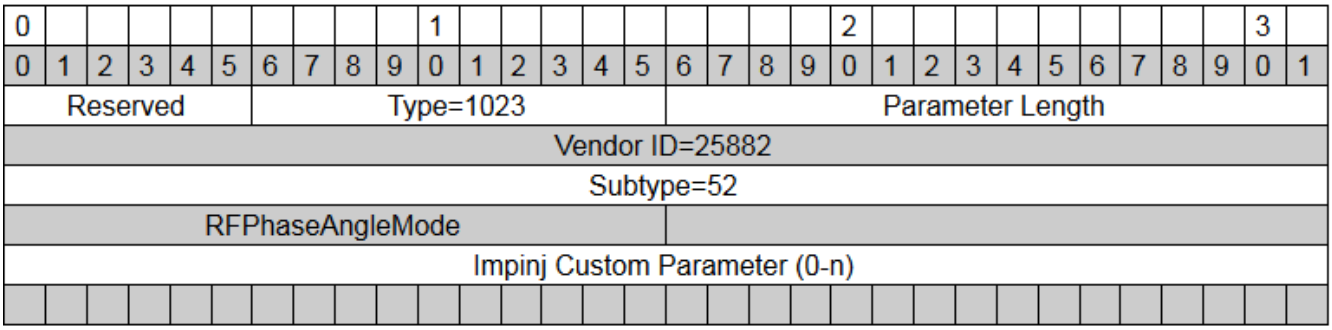

For more information, see section 4.3.25.

### 6.2.28 ImpinjEnableSerializedTID Parameter



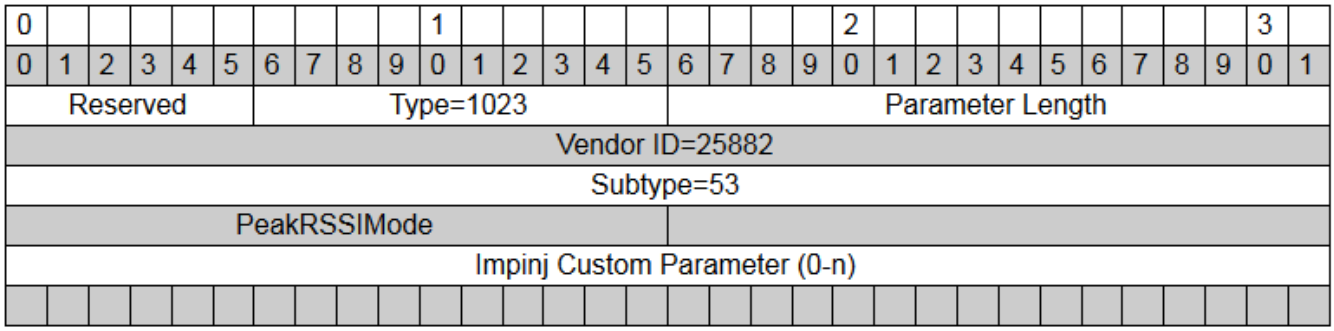

See section 4.3.26.

### 6.2.29 ImpinjEnableRFPhaseAngle Parameter



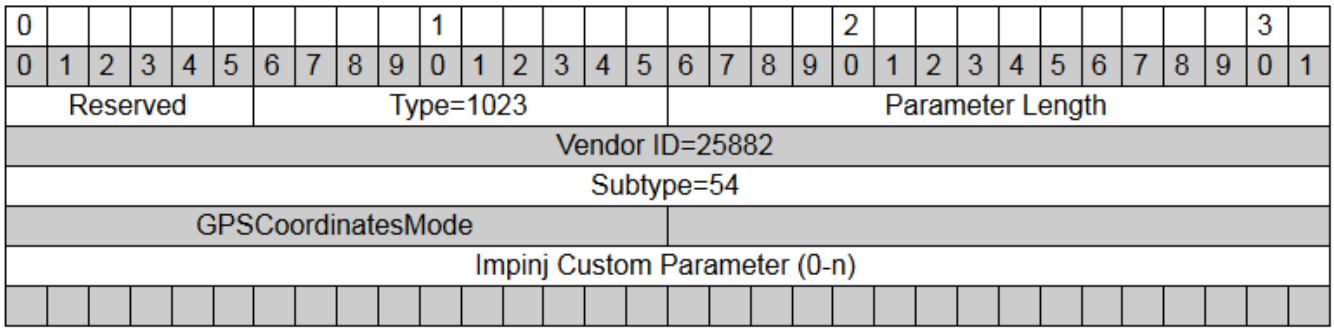

For more information, see section 4.3.27.

### 6.2.30 ImpinjEnablePeakRSSI Parameter



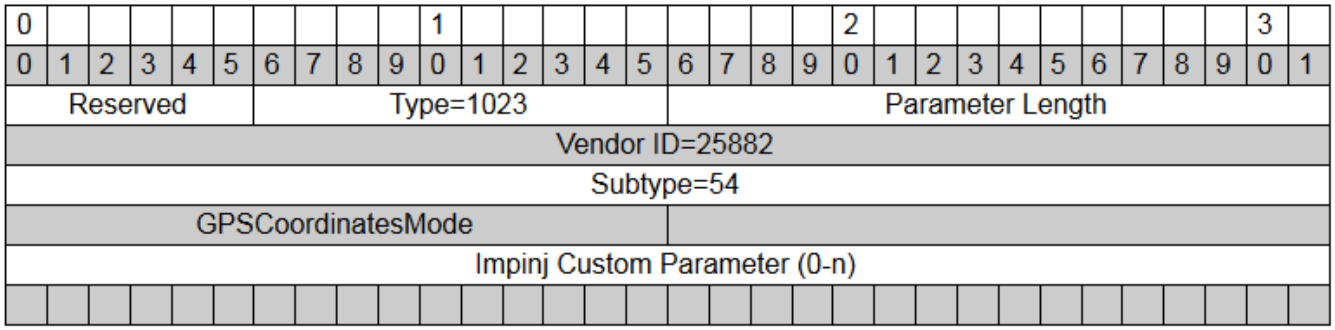

For more information, see section 4.3.28.

### 6.2.31 ImpinjEnableGPSCoordinates Parameter



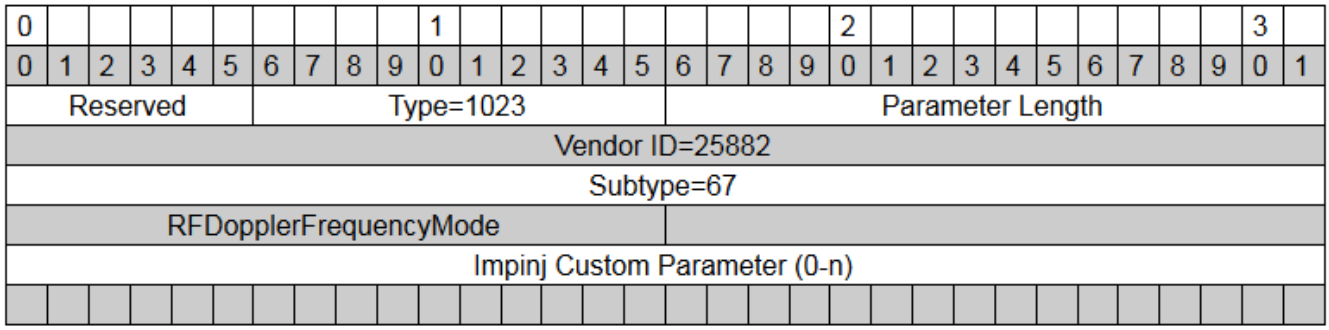

For more information, see section 4.3.29.

### 6.2.32 ImpinjEnableOptimizedRead Parameter



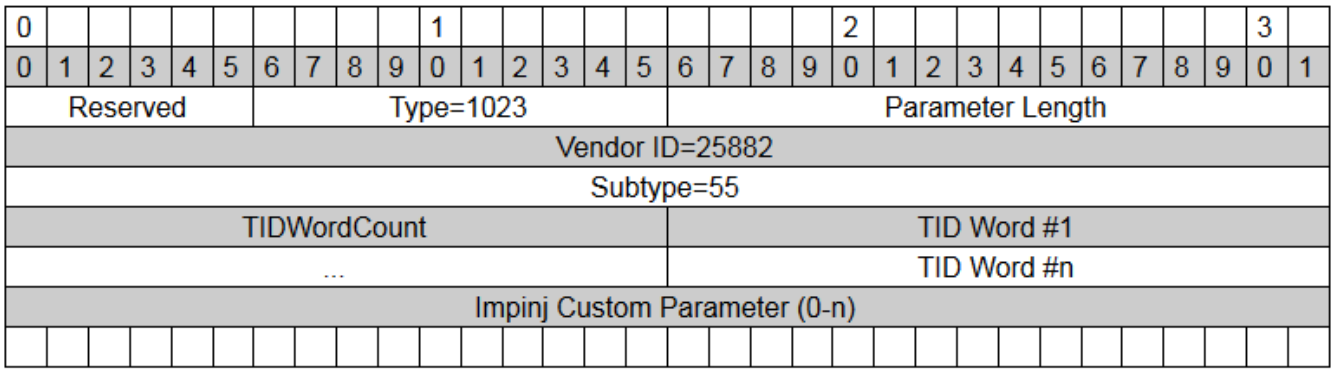

For more information, see section 4.3.30.

### 6.2.33 ImpinjEnableRFDopplerFrequency Parameter



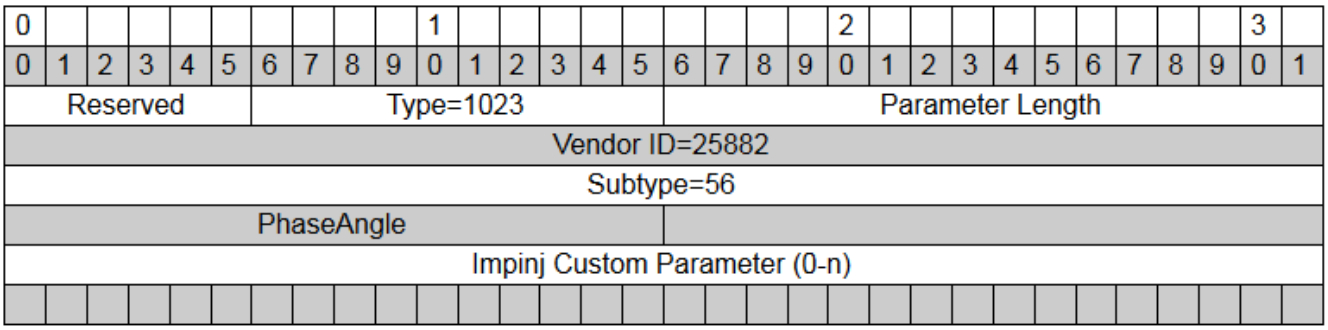

For more information, see section 4.3.31.

### 6.2.34 ImpinjSerializedTID Parameter



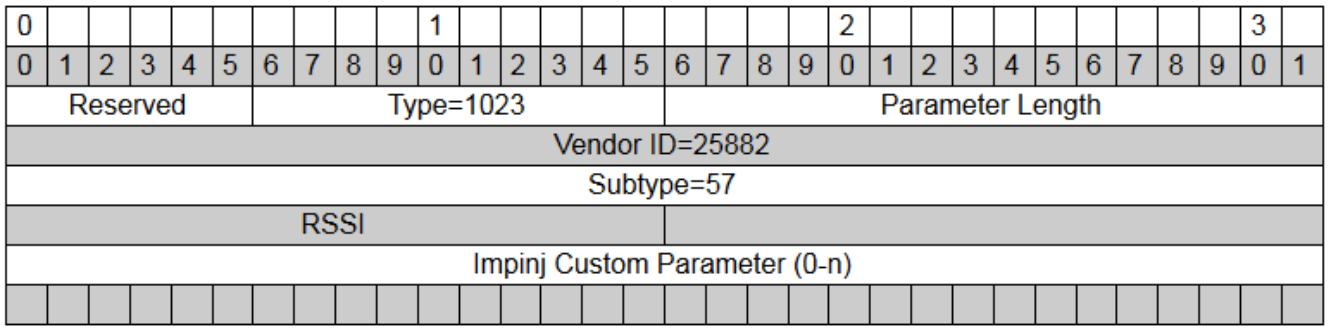

For more information, see section 4.3.32.

### 6.2.35 ImpinjRFPhaseAngle Parameter



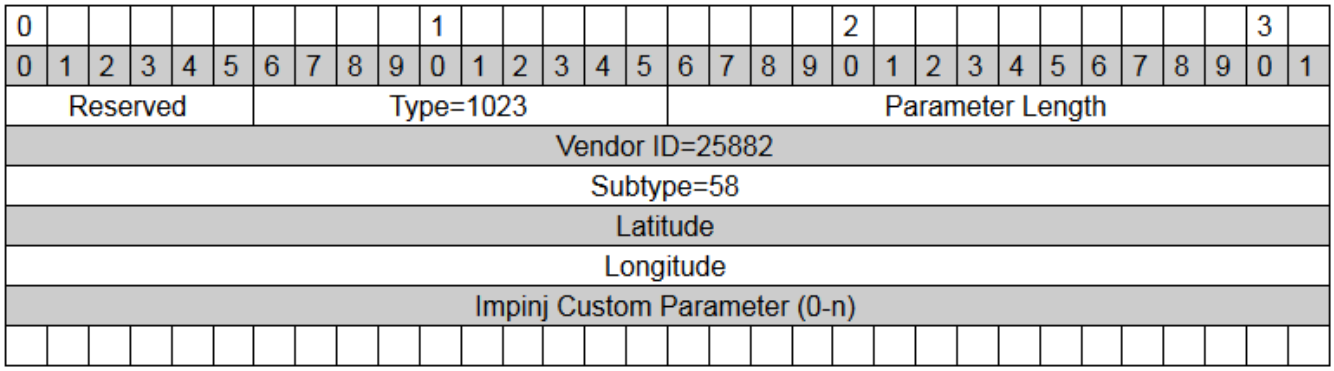

For more information, see section 4.3.33.

### 6.2.36 ImpinjPeakRSSI Parameter



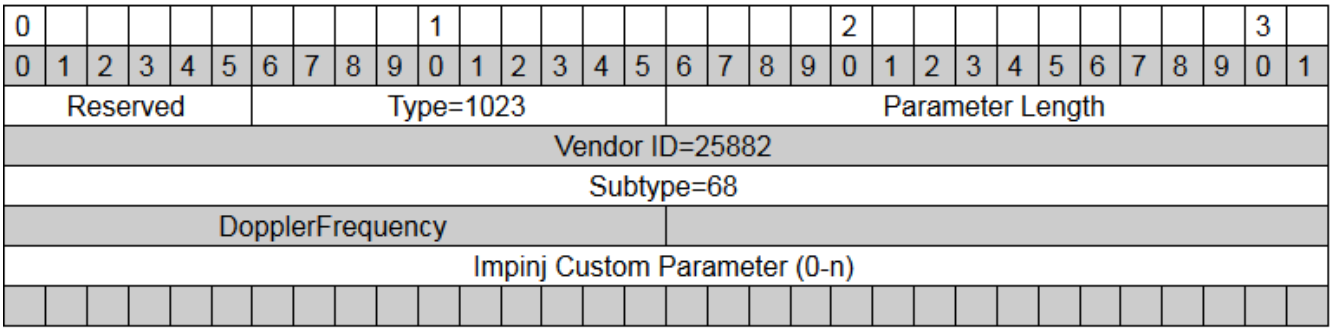

For more information, see section 4.3.34.

### 6.2.37 ImpinjGPSCoordinates Parameter



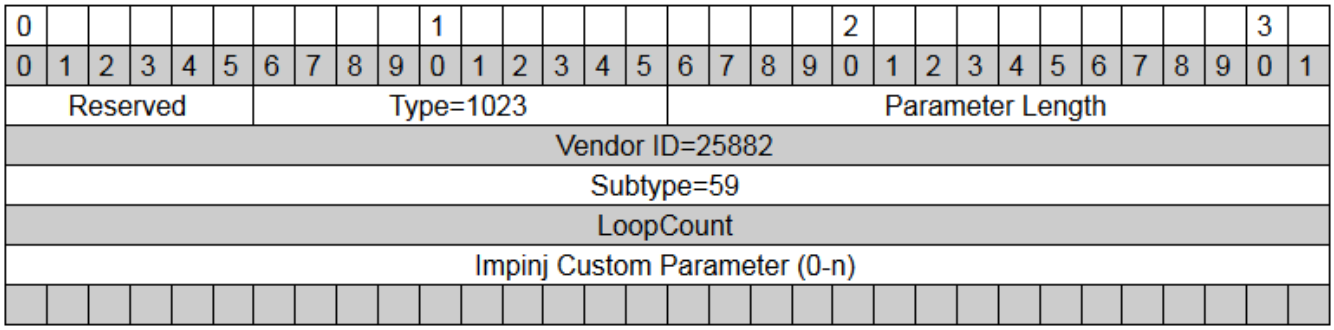

For more information, see section 4.3.35.

### 6.2.38 ImpinjRFDopplerFrequency Parameter



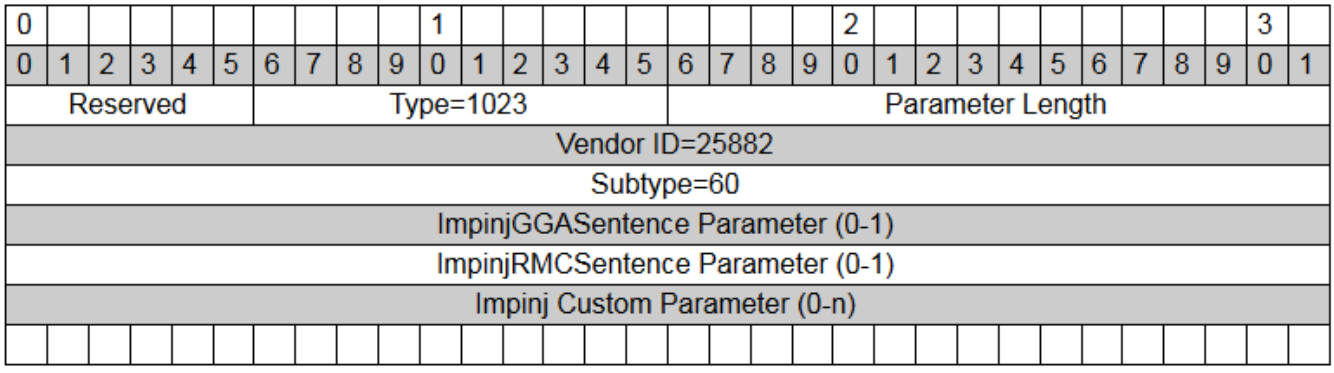

For more information, see section 4.3.36.

### 6.2.39 ImpinjLoopSpec Parameter



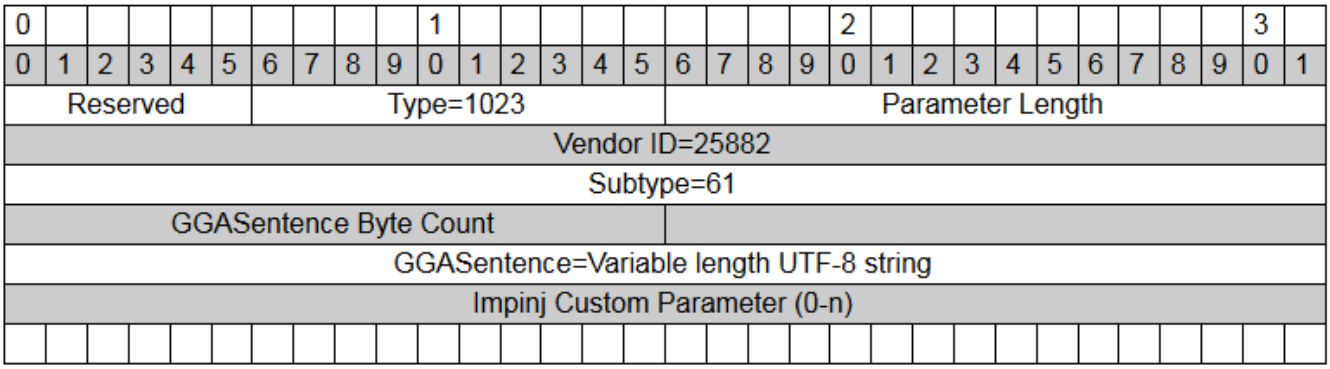

For more information, see section 4.3.37.

### 6.2.40 ImpinjGPSNMEASentences Parameter



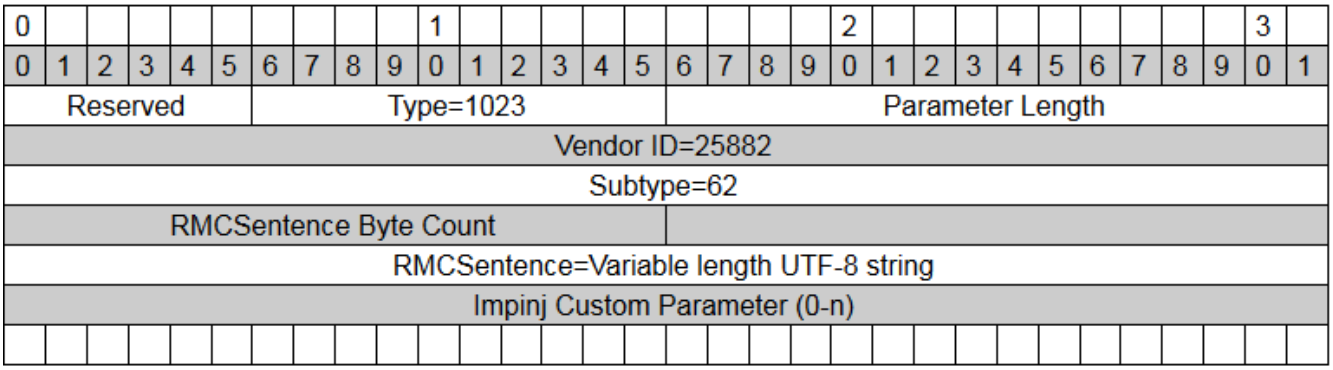

For more information, see section 4.3.38.

### 6.2.41 ImpinjGGASentence Parameter



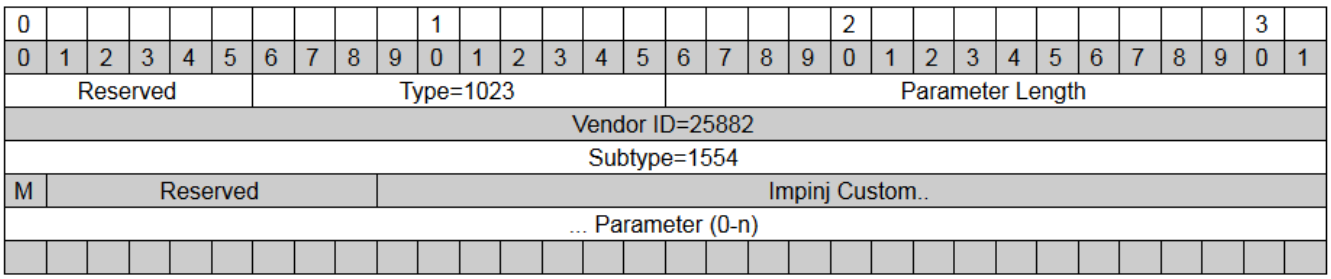

For more information, see section 4.3.39.

### 6.2.42 ImpinjRMCSentence Parameter




For more information, see section 4.3.40.

### 6.2.43 ImpinjIntelligentAntennaManagement Parameter




Abbreviations: M - ManagementEnabled

For more information, see section 4.3.41.

## 6.3 xArray Parameters

The following parameters only pertain to the xArray Gateway.

### 6.3.1 ImpinjArrayVersion Parameter

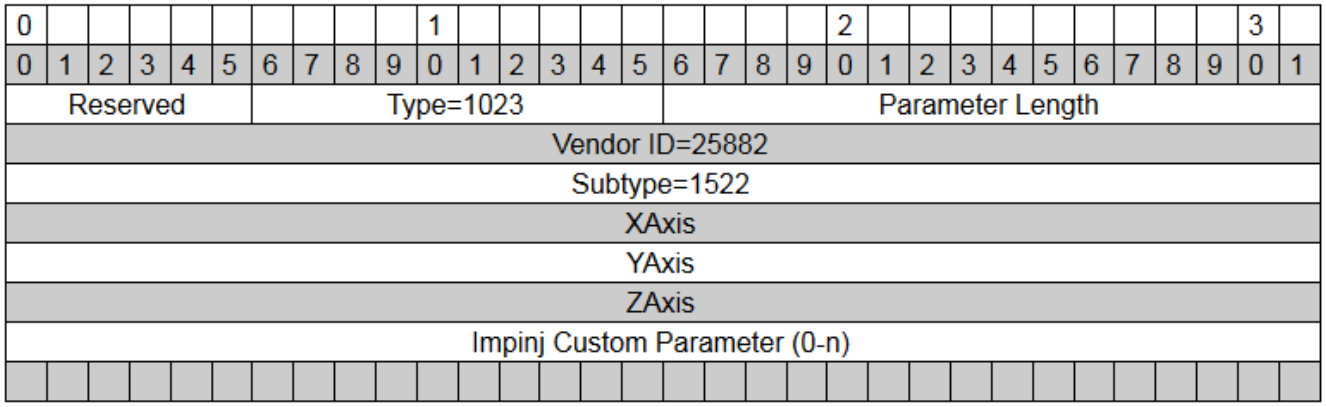
| 0 | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | 3 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| Reserved | | | | | | Type=1023 | | | | | | | | | | Parameter Length | | | | | | | | | | | | | | |
| Vendor ID=25882 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Subtype=1520 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| SerialNumber Byte Count | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| SerialNumber=Variable length UTF-8 string | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| FirmwareVersion Byte Count | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| FirmwareVersion=Variable length UTF-8 string | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| PCBAVersion Byte Count | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| PCBAVersion=Variable length UTF-8 string | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Impinj Custom Parameter (0-n) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

### 6.3.2 ImpinjxArrayCapabilities Parameter

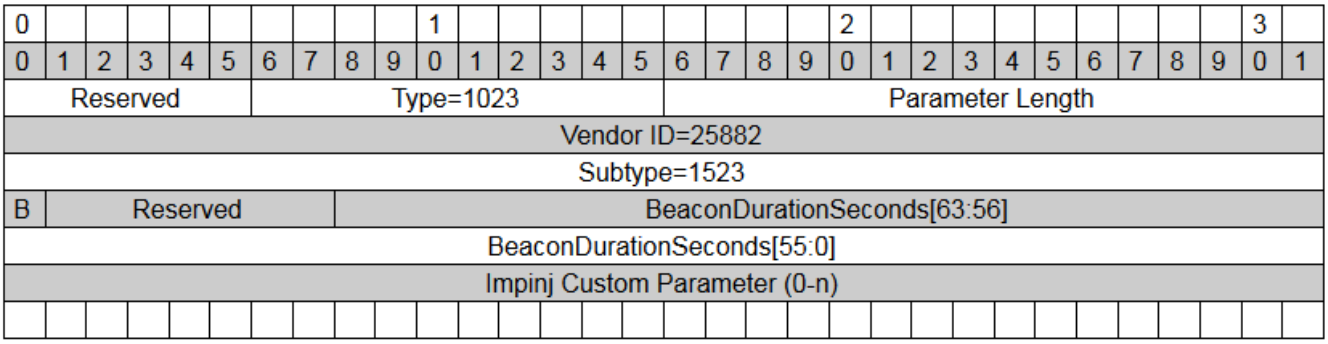
| 0 | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | 3 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| Reserved | | | | | | Type=1023 | | | | | | | | | | Parameter Length | | | | | | | | | | | | | | |
| Vendor ID=25882 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Subtype=1521 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| MaxNumSectors[31:16] | | | | | | | | | | | | | | | | MaxNumSectors[15:0] | | | | | | | | | | | | | | |
| L | T | Reserved | | | | | | | | | | | | | | Impinj Custom Parameter (0-n) | | | | | | | | | | | | | | |

Abbreviations: L - SupportsLISpecs

### 6.3.3 ImpinjTiltConfiguration Parameter

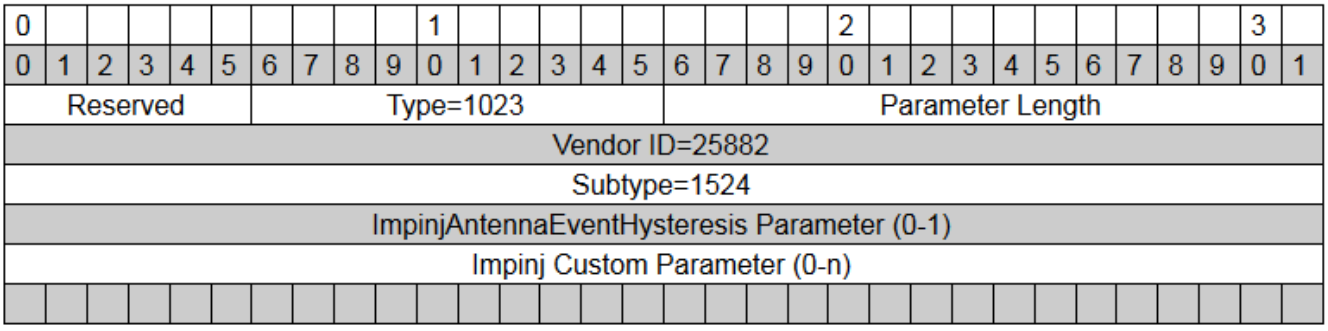



### 6.3.4 ImpinjBeaconConfiguration Parameter

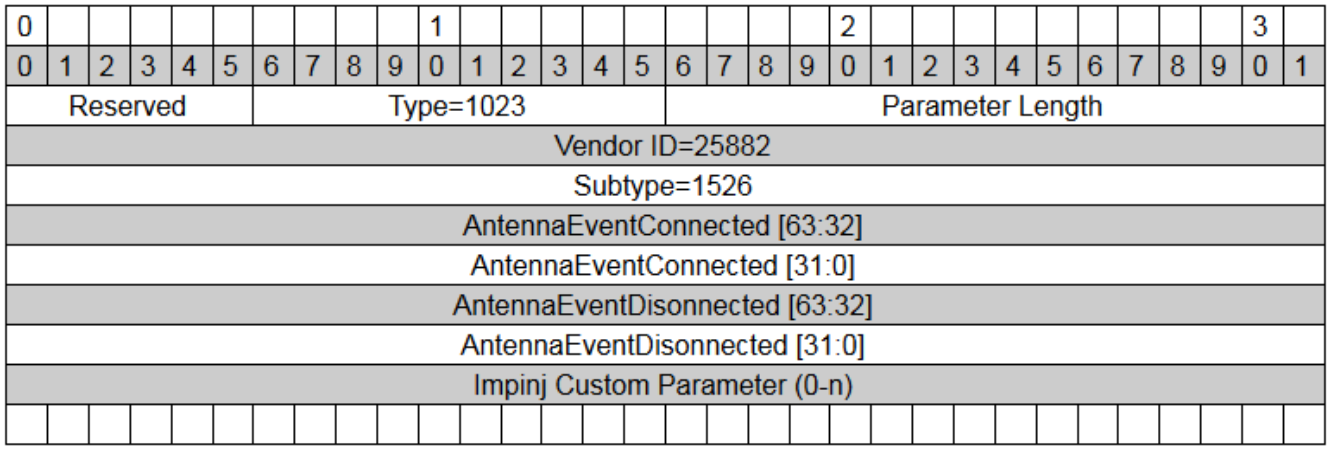



Abbreviations: B - BeaconState

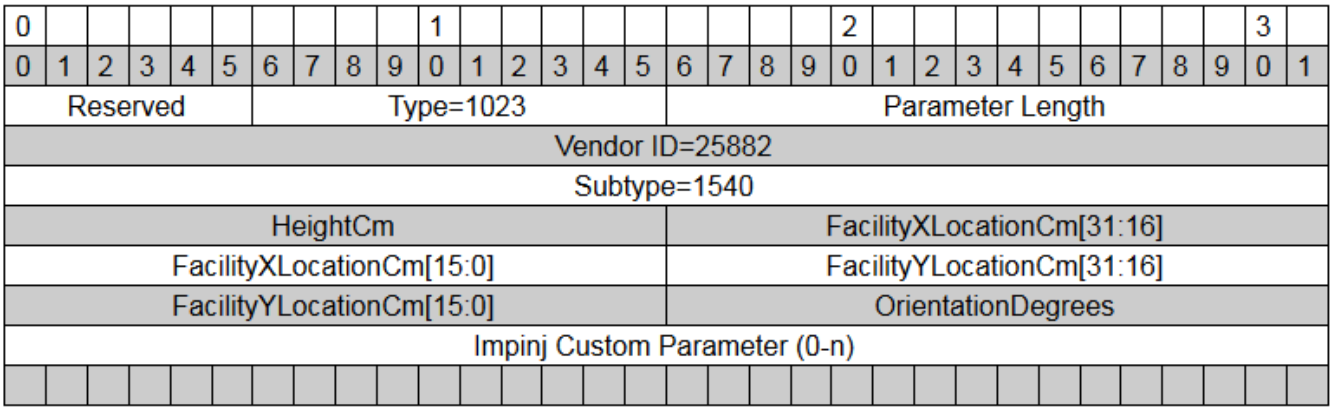
### 6.3.5 ImpinjAntennaConfiguration Parameter



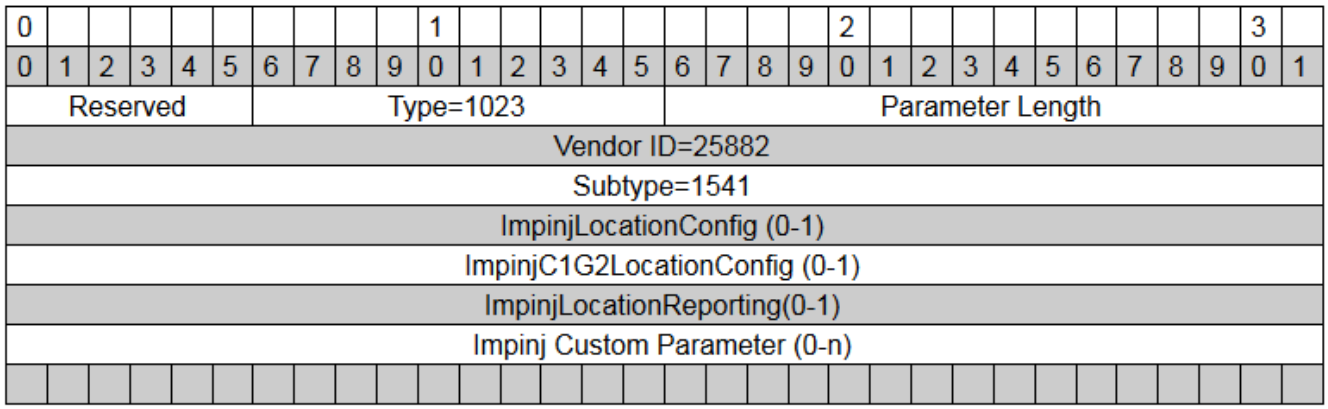
### 6.3.6 ImpinjAntennaEventHysteresis Parameter



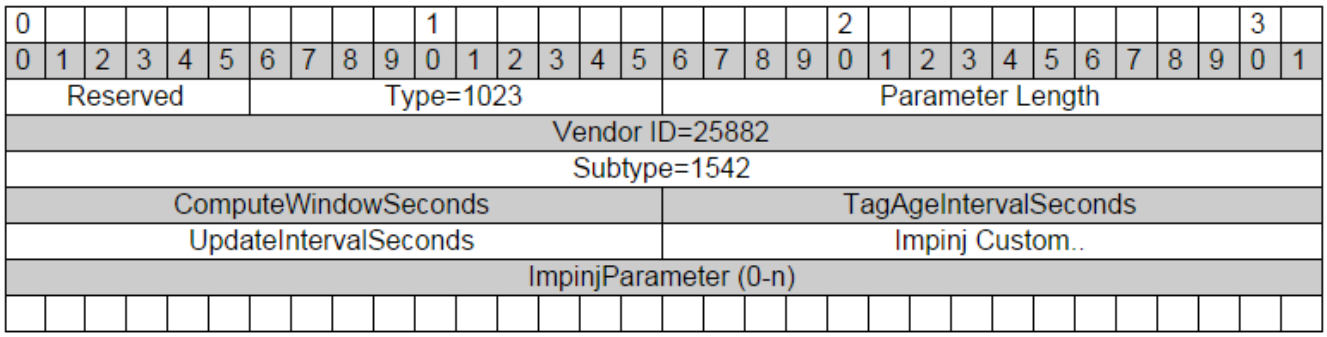

### 6.3.7 ImpinjPlacementConfiguration Parameter



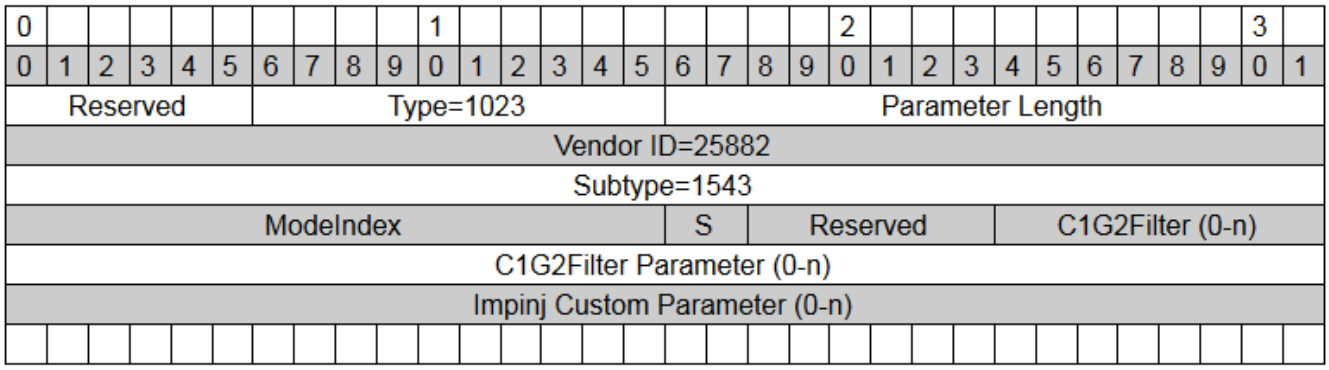
### 6.3.8 ImpinjLISpec




### 6.3.9 ImpinjLocationConfig Parameter

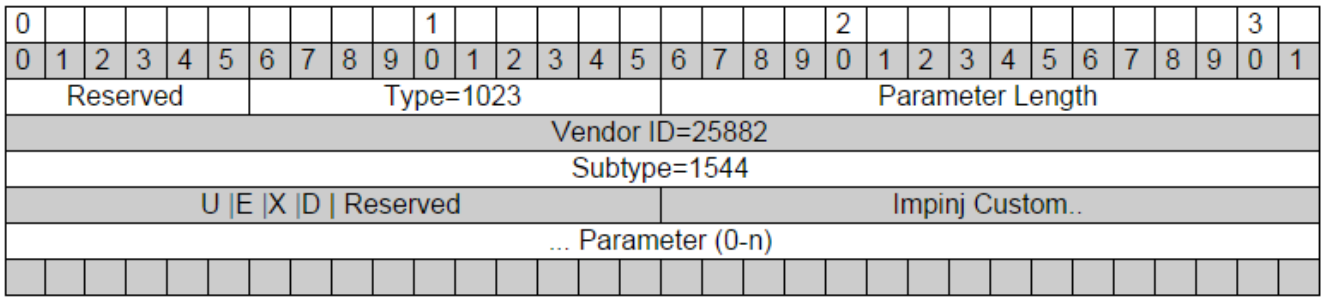



### 6.3.10 ImpinjC1G2LocationConfig



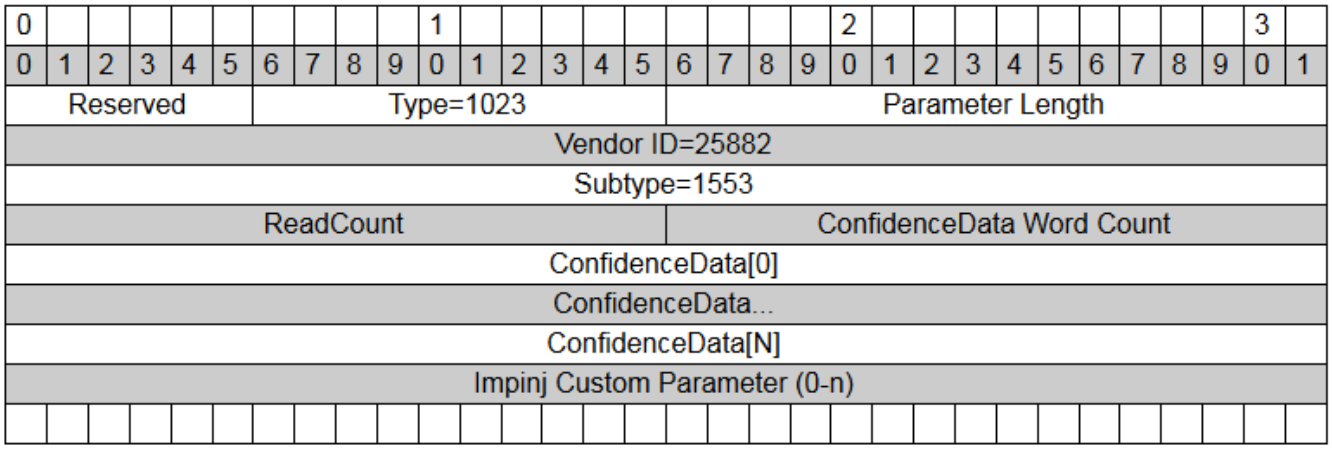

Abbreviations: S - Session

### 6.3.11 ImpinjLocationReporting Parameter

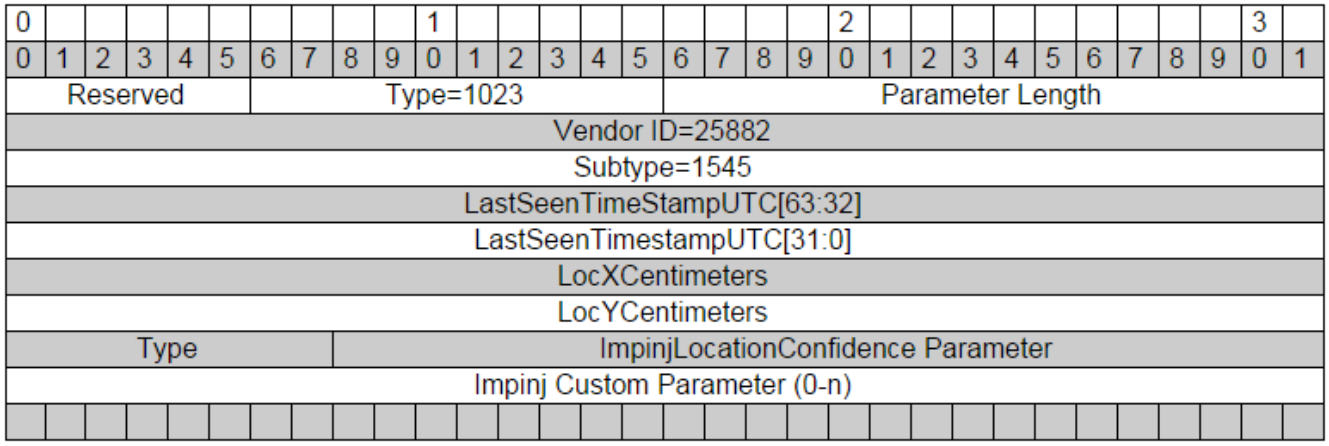



**Abbreviations:** U - EnableUpdateReport E - EnableEntryReport X - EnableExitReport D - EnableDiagnosticReport

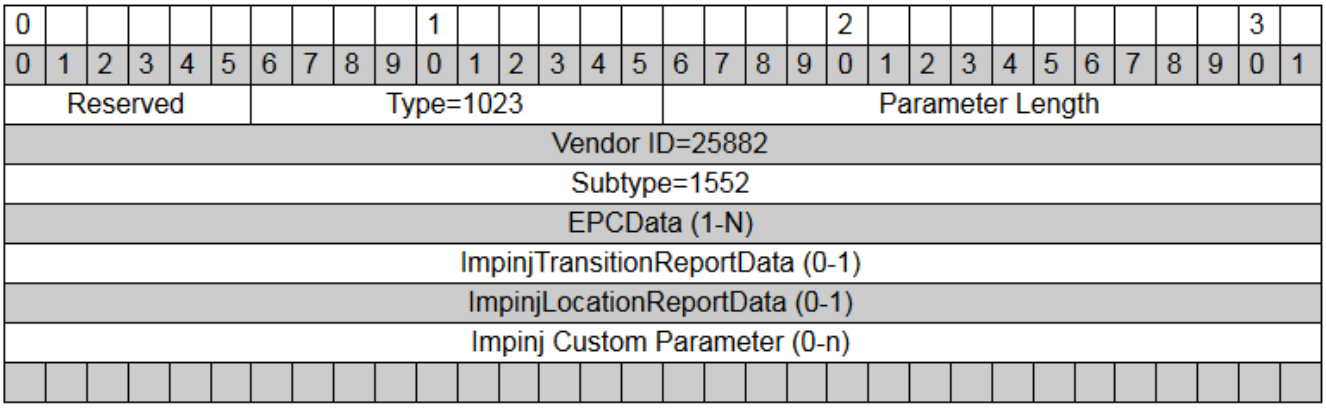
### 6.3.12 ImpinjLocationConfidence Parameter



### 6.3.13 ImpinjLocationReportData Parameter




### 6.3.14 ImpinjExtendedTagInformation Parameter



# 7 Octane LLRP Toolkit Information

Octane LLRP is tested against some libraries that are produced by the open source llrp-toolkit (LTK) project. Table 7.1 shows the compatibility of Octane 5.2 with the LTK. For information about building custom versions of the LTK, go to the toolkit site: 3TU http://www.sourceforge.net/projects/llrp-toolkitU3T.

**Table 7.1 Octane LTK Compatibility**

| Language | Version | Availability | Notes |
|---|---|---|---|
| Perl | 1.0.x | Sourceforge | Available as open source; not fully tested against Speedway, Speedway Revolution, xArray Gateway |
| C | 10.18.1.240 | Impinj | |
| C++ | 10.18.1.240 | Impinj | |
| C# .NET | 10.18.1.240 | Impinj | |
| Java | 10.18.1.240 | Impinj | |

# 8 Octane LLRP Default Values

The following table describes the factory default values for LLRP and Octane custom extension parameters for the available Octane regulatory regions. Commanding the Reader to restore LLRP factory defaults via the *ResetToFactoryDefault* field of the LLRP SET_READER_CONFIG message will restore the Reader to these factory defaults. Non-LLRP settings such as network settings, root password, or other settings programmable via a separate API on the Reader are not affected.

**Table 8.1 Octane LLRP Default Configuration Values (1)**

| | FCC | ETSI | Hong Kong | Taiwan | Japan | Korea | Malaysia | China |
|---|---|---|---|---|---|---|---|---|
| **Channel Index** | N/A | Index 3<br>Channel 10<br>866.9 MHz | N/A | N/A | Index 1<br>Channel 2<br>952.4 MHz | N/A | N/A | Index 1<br>Channel 3<br>920.625 MHz |
| **Hop Table ID** | 1 | N/A | 1 | 1 | N/A | 1 | 1 | N/A |
| **Transmit Power** | 30 dBm (:under`TransmitPower` index varies by product) | | | | | | | |
| **Receive Sensitivity** | 0 dB above maximum sensitivity | | | | | | | |
| **Gen2 Mode ID** | 1000 (Autoset) | | | | | | | |
| **Session** | Session 1 | | | | | | | |
| **Tag Transit Time** | 0 | | | | | | | |
| **Tag Population** | 32 | | | | | | | |
| **RO Spec** | No ROSpecs are configured | | | | | | | |
| **Access Spec** | No AccessSpecs are configured | | | | | | | |
| **RO Report Trigger** | Tag data will be reported on each singulation (N=1) | | | | | | | |
| **Tag Report Data** | AntennaID, PeakRSSI, and FirstSeenTimestamp are enabled | | | | | | | |
| **Access Report Trigger** | Access data will be reported at the completion of each AccessSpec | | | | | | | |
| **Keep Alive** | Disabled | | | | | | | |
| **Reader Events** | Antenna and ReaderExceptionEvents are enabled | | | | | | | |
| **Hold Events and Reports** | False | | | | | | | |
| **GPI Configuration** | All GPI are disabled | | | | | | | |
| **GPI Debounce Timer** | 20 milliseconds | | | | | | | |
| **GPO State** | All GPO are driven low | | | | | | | |
| **Impinj Extensions** | Disabled | | | | | | | |
| **Impinj Sub-Regulatory Region** | 0 | 7 | 3 | 4 | 11 | 8 | 9 | 10 |
| **Impinj Fixed Frequency List** | Disabled | | | | | | | |
| **Impinj Low Duty Cycle** | Disabled | | | | | | | |
| **Impinj Reduced Power Frequency List** | Disabled | | | | | | | |
| **Impinj Inventory Search Mode** | Reader will select the appropriate search mode | | | | | | | |
| **Link State Monitoring** | Disabled | | | | | | | |
| **Report Buffer Behavior** | Normal | | | | | | | |
| **Block Write Word Count** | 1 | | | | | | | |
| **Serialized TID Reporting** | Disabled | | | | | | | |
| **RF Phase Reporting** | Disabled | | | | | | | |
| **Intelligent Antenna Management** | Enabled | | | | | | | |

**Table 8.2 Octane LLRP Default Configuration Values (2)**

| | South Africa | Brazil | Thailand | Singapore | Australia | India | Uruguay | Vietnam | Israel |
|---|---|---|---|---|---|---|---|---|---|
| **Channel Index** | N/A | N/A | N/A | N/A | N/A | Index 3<br>Channel 10<br>866.9 MHz | N/A | N/A | N/A |
| **Hop Table ID** | 1 | 1 | 1 | 1 | 1 | N/A | 1 | 1 | 1 |
| **Transmit Power** | 30 dBm (UTransmitPowerU index varies by product) | | | | | | | | |
| **Receive Sensitivity** | 0 dB above maximum sensitivity | | | | | | | | |
| **Gen2 Mode ID** | 1000 (Autoset) | | | | | | | | |
| **Session** | Session 1 | | | | | | | | |
| **Tag Transit Time** | 0 | | | | | | | | |
| **Tag Population** | 32 | | | | | | | | |
| **RO Spec** | No ROSpecs are configured | | | | | | | | |
| **Access Spec** | No AccessSpecs are configured | | | | | | | | |
| **RO Report Trigger** | Tag data will be reported on each singulation (N=1) | | | | | | | | |
| **Tag Report Data** | AntennaID, PeakRSSI, and FirstSeenTimestamp are enabled | | | | | | | | |
| **Access Report Trigger** | Access data will be reported at the completion of each AccessSpec | | | | | | | | |
| **Keep Alive** | Disabled | | | | | | | | |
| **Reader Events** | Antenna and ReaderExceptionEvents are enabled | | | | | | | | |
| **Hold Events and Reports** | False | | | | | | | | |
| **GPI Configuration** | All GPI are disabled | | | | | | | | |
| **GPI Debounce Timer** | 20 milliseconds | | | | | | | | |
| **GPO State** | All GPO are driven low | | | | | | | | |
| **Impinj Extensions** | Disabled | | | | | | | | |
| **Impinj Sub-Regulatory Region** | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| **Impinj Fixed Frequency List** | Disabled | | | | | | | | |
| **Impinj Low Duty Cycle** | Disabled | | | | | | | | |
| **Impinj Reduced Power Frequency List** | Disabled | | | | | | | | |
| **Impinj Inventory Search Mode** | Reader will select the appropriate search mode | | | | | | | | |
| **Link State Monitoring** | Disabled | | | | | | | | |
| **Report Buffer Behavior** | Normal | | | | | | | | |
| **Block Write Word Count** | 1 | | | | | | | | |
| **Serialized TID Reporting** | Disabled | | | | | | | | |
| **RF Phase Reporting** | Disabled | | | | | | | | |
| **Intelligent Antenna Management** | Enabled | | | | | | | | |

**Table 8.3 Octane LLRP Default Configuration Values (3)**

| | Philippines | Canada Post | Indonesia | New Zealand | Japan | Latin America | Peru |
|---|---|---|---|---|---|---|---|
| **Channel Index** | Index 3<br>Channel 1<br>918.25 MHz | N/A | N/A | N/A | Index 3<br>Channel 18<br>919.2 MHz | N/A | N/A |
| **Hop Table ID** | N/A | 1 | 1 | 1 | | 1 | 1 |
| **Transmit Power** | 30 dBm (TransmitPower index varies by product) | | | | | | |
| **Receive Sensitivity** | 0 dB above maximum sensitivity | | | | | | |
| **Gen2 Mode ID** | 1000 (Autoset) | | | | | | |
| **Session** | Session 1 | | | | | | |
| **Tag Transit Time** | 0 | | | | | | |
| **Tag Population** | 32 | | | | | | |
| **RO Spec** | No ROSpecs are configured | | | | | | |
| **Access Spec** | No AccessSpecs are configured | | | | | | |
| **RO Report Trigger** | Tag data will be reported on each singulation (N=1) | | | | | | |
| **Tag Report Data** | AntennaID, PeakRSSI, and FirstSeenTimestamp are enabled | | | | | | |
| **Access Report Trigger** | Access data will be reported at the completion of each AccessSpec | | | | | | |
| **Keep Alive** | Disabled | | | | | | |
| **Reader Events** | Antenna and ReaderExceptionEvents are enabled | | | | | | |
| **Hold Events and Reports** | False | | | | | | |
| **GPI Configuration** | All GPI are disabled | | | | | | |
| **GPI Debounce Timer** | 20 milliseconds | | | | | | |
| **GPO State** | All GPO are driven low | | | | | | |
| **Impinj Extensions** | Disabled | | | | | | |
| **Impinj Sub-Regulatory Region** | 21 | 22 | 23 | 24 | 25 | 26 | 27 |
| **Impinj Fixed Frequency List** | Disabled | | | | | | |
| **Impinj Low Duty Cycle** | Disabled | | | | | | |
| **Impinj Reduced Power Frequency List** | Disabled | | | | | | |
| **Impinj Inventory Search Mode** | Reader will select the appropriate search mode | | | | | | |
| **Link State Monitoring** | Disabled | | | | | | |
| **Report Buffer Behavior** | Normal | | | | | | |
| **Block Write Word Count** | 1 | | | | | | |
| **Serialized TID Reporting** | Disabled | | | | | | |
| **RF Phase Reporting** | Disabled | | | | | | |
| **Intelligent Antenna Management** | Enabled | | | | | | |

# 9 Regulatory Region Information

The tables in this section provide information about the capabilities of the Octane firmware in each regulatory region. This is for information only. For an accurate and complete list of a Reader's capabilities in a particular regulatory region, use the LLRP GET_READER_CAPABILITIES message.

**xArray Gateway** Unlike the Speedway Revolution Reader, transmit power is automatically controlled by the xArray Gateway.

- Table 9.1 documents the information contained in the *TransmitPowerLevelTableEntry* parameter list within *UHFBandCapabilities*.
- Table 9.2 documents the information contained in the *FrequencyList* field within *ImpinjFrequencyCapabilities*.
- Table 9.3 documents the information contained in the *ReceiveSensitivityTableEntry* parameter list within *GeneralDeviceCapabilities*.

## 9.1 Table 9.1 Regional Transmit Power Capabilities (dBm)

| Region | R220 [1] | R420 [1] | R640 [2] |
|---|---|---|---|
| 0: FCC part 15.247 | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 1: ETSI EN 300-220 | N/A | N/A | N/A |
| 2: ETSI EN 302-208 | See region 7 | See region 7 | See region 7 |
| 3: Hong Kong 920-925 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 4: Taiwan 922-928 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 5: Japan 952-954 MHz | N/A | N/A | N/A |
| 6: Japan 952-955 MHz, 10mW max power | N/A | N/A | N/A |
| 7: ETSI EN 302-208 (version 1.4.1) | 10.00 - 31.50 | 10.00 - 31.50 | 10.00 - 30.00 |
| 8: Korea 917-921 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 9: Malaysia 919-923MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 10: China 920-925 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 11: Japan 952-956 MHz (without LBT) | 10.00 - 30.00 | 10.00 - 30.00 | N/A |
| 12: South Africa 915-919 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 13: Brazil 902-907/915-928 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 14: Thailand 920-925 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 15: Singapore 920-925 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |

| Region | R220 [1] | R420 [1] | R640 [2] |
|---|---|---|---|
| 16: Australia 920-926 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 17: India 865-867 MHz | 10.00 - 31.50 | 10.00 - 31.50 | 10.00 - 30.00 |
| 18: Uruguay 916-928 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 19: Vietnam 920-925 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 20: Israel 915-917 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 21: Philippines 918-920 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 22: Canada Post 902-928 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 23: Indonesia 923-925 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 24: New Zealand 921.5-928 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 25: Japan 916.7-920.9 MHz | N/A | 10.00-30.00 | N/A |
| 26: Latin America 902-928 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |
| 27: Peru 916-928 MHz | 10.00 - 32.50 | 10.00 - 32.50 | 10.00 - 30.00 |

[1] R220 and R420 - only supports power above 30.0 dBm when powered via an external power supply. Power is limited to 30.0 dBm when powered via PoE.

[2] xPortal R640 - the user specifies transmit power in the 10 to 30 dBm range. The Octane firmware will limit the transmit power as required by the regulatory region given the antenna gain of 7.5 dBm.

## 9.2 Table 9.2 Regional Frequency Capabilities

| Region | ChannelIndex | Frequency (MHz) |
|---|---|---|
| 0: FCC part 15.247 | 1 | 902.750 |
| | 2 | 903.250 |
| | 3 | 903.750 |
| | 4 | 904.250 |
| | 5 | 904.750 |
| | 6 | 905.250 |
| | 7 | 905.750 |
| | 8 | 906.250 |
| | 9 | 906.750 |
| | 10 | 907.250 |
| | 11 | 907.750 |
| | 12 | 908.250 |

| Region | ChannelIndex | Frequency (MHz) |
|---|---|---|
| | 13 | 908.750 |
| | 14 | 909.250 |
| | 15 | 909.750 |
| | 16 | 910.250 |
| | 17 | 910.750 |
| | 18 | 911.250 |
| | 19 | 911.750 |
| | 20 | 912.250 |
| | 21 | 912.750 |
| | 22 | 913.250 |
| | 23 | 913.750 |
| | 24 | 914.250 |
| | 25 | 914.750 |
| | 26 | 915.250 |
| | 27 | 915.750 |
| | 28 | 916.250 |
| | 29 | 916.750 |
| | 30 | 917.250 |
| | 31 | 917.750 |
| | 32 | 918.250 |
| | 33 | 918.750 |
| | 34 | 919.250 |
| | 35 | 919.750 |
| | 36 | 920.250 |
| | 37 | 920.750 |
| | 38 | 921.250 |
| | 39 | 921.750 |
| | 40 | 922.250 |
| | 41 | 922.750 |
| | 42 | 923.250 |
| | 43 | 923.750 |
| | 44 | 924.250 |
| | 45 | 924.750 |
| | 46 | 925.250 |
| | 47 | 925.750 |
| | 48 | 926.250 |
| | 49 | 926.750 |
| | 50 | 927.250 |
| 1: ETSI EN 300-220 | N/A | N/A |

| Region | ChannelIndex | Frequency (MHz) |
|---|---|---|
| 2: ETSI EN 302-208 | See region 7 | See region 7 |
| 3: Hong Kong 920-925 MHz | 1 | 920.250 |
| | 2 | 920.750 |
| | 3 | 921.250 |
| | 4 | 921.750 |
| | 5 | 922.250 |
| | 6 | 922.750 |
| | 7 | 923.250 |
| | 8 | 923.750 |
| | 9 | 924.250 |
| | 10 | 924.750 |
| 4: Taiwan 922-928 MHz | 1 | 922.250 |
| | 2 | 922.750 |
| | 3 | 923.250 |
| | 4 | 923.750 |
| | 5 | 924.250 |
| | 6 | 924.750 |
| | 7 | 925.250 |
| | 8 | 925.750 |
| | 9 | 926.250 |
| | 10 | 926.750 |
| | 11 | 927.250 |
| | 12 | 927.750 |
| 5: Japan 952-954 MHz | N/A | N/A |
| 6: Japan 952-955 MHz, 10mW max power | N/A | N/A |
| 7: ETSI EN 302-208 (version 1.4.1) | 1 | 865.700 |
| | 2 | 866.300 |
| | 3 | 866.900 |
| | 4 | 867.500 |
| 8: Korea 917-921 MHz | 1 | 917.300 |
| | 2 | 917.900 |
| | 3 | 918.500 |
| | 4 | 919.100 |
| | 5 | 919.700 |
| | 6 | 920.300 |
| 9: Malaysia 919-923MHz | 1 | 919.250 |
| | 2 | 919.750 |

| Region | ChannelIndex | Frequency (MHz) |
|---|---|---|
| | 3 | 920.250 |
| | 4 | 920.750 |
| | 5 | 921.250 |
| | 6 | 921.750 |
| | 7 | 922.250 |
| | 8 | 922.750 |
| 10: China 920-925 MHz | 1 | 920.625 |
| | 2 | 920.875 |
| | 3 | 921.125 |
| | 4 | 921.375 |
| | 5 | 921.625 |
| | 6 | 921.875 |
| | 7 | 922.125 |
| | 8 | 922.375 |
| | 9 | 922.625 |
| | 10 | 922.875 |
| | 11 | 923.125 |
| | 12 | 923.375 |
| | 13 | 923.625 |
| | 14 | 923.875 |
| | 15 | 924.125 |
| | 16 | 924.375 |
| 11: Japan 952-956 MHz (without LBT) | 1 | 952.400 |
| | 2 | 953.600 |
| | 3 | 954.800 |
| | 4 | 956.000 |
| 12: South Africa 915-919 MHz | 1 | 915.600 |
| | 2 | 915.800 |
| | 3 | 916.000 |
| | 4 | 916.200 |
| | 5 | 916.400 |
| | 6 | 916.600 |
| | 7 | 916.800 |
| | 8 | 917.000 |
| | 9 | 917.200 |
| | 10 | 917.400 |
| | 11 | 917.600 |
| | 12 | 917.800 |

| Region | ChannelIndex | Frequency (MHz) |
|---|---|---|
| | 13 | 918.000 |
| | 14 | 918.200 |
| | 15 | 918.400 |
| | 16 | 918.600 |
| | 17 | 918.800 |
| 13: Brazil 902-907/915-928 MHz | 1 | 902.750 |
| | 2 | 903.250 |
| | 3 | 903.750 |
| | 4 | 904.250 |
| | 5 | 904.750 |
| | 6 | 905.250 |
| | 7 | 905.750 |
| | 8 | 906.250 |
| | 9 | 906.750 |
| | 10 | 907.250 |
| | 11 | 915.250 |
| | 12 | 915.750 |
| | 13 | 916.250 |
| | 14 | 916.750 |
| | 15 | 917.250 |
| | 16 | 917.750 |
| | 17 | 918.250 |
| | 18 | 918.750 |
| | 19 | 919.250 |
| | 20 | 919.750 |
| | 21 | 920.250 |
| | 22 | 920.750 |
| | 23 | 921.250 |
| | 24 | 921.750 |
| | 25 | 922.250 |
| | 26 | 922.750 |
| | 27 | 923.250 |
| | 28 | 923.750 |
| | 29 | 924.250 |
| | 30 | 924.750 |
| | 31 | 925.250 |
| | 32 | 925.750 |
| | 33 | 926.250 |
| | 34 | 926.750 |

| Region | ChannelIndex | Frequency (MHz) |
|---|---|---|
| | 35 | 927.250 |
| 14: Thailand 920-925 MHz | 1 | 920.250 |
| | 2 | 920.750 |
| | 3 | 921.250 |
| | 4 | 921.750 |
| | 5 | 922.250 |
| | 6 | 922.750 |
| | 7 | 923.250 |
| | 8 | 923.750 |
| | 9 | 924.250 |
| | 10 | 924.750 |
| 15: Singapore 920-925 MHz | 1 | 920.250 |
| | 2 | 920.750 |
| | 3 | 921.250 |
| | 4 | 921.750 |
| | 5 | 922.250 |
| | 6 | 922.750 |
| | 7 | 923.250 |
| | 8 | 923.750 |
| | 9 | 924.250 |
| | 10 | 924.750 |
| 16: Australia 920-926 MHz | 1 | 920.250 |
| | 2 | 920.750 |
| | 3 | 921.250 |
| | 4 | 921.750 |
| | 5 | 922.250 |
| | 6 | 922.750 |
| | 7 | 923.250 |
| | 8 | 923.750 |
| | 9 | 924.250 |
| | 10 | 924.750 |
| | 11 | 925.250 |
| | 12 | 925.750 |
| 17: India 865-867 MHz | 1 | 865.100 |
| | 2 | 865.700 |
| | 3 | 866.300 |
| | 4 | 866.900 |
| 18: Uruguay 916-928 MHz | 1 | 916.250 |
| | 2 | 916.750 |

| Region | ChannelIndex | Frequency (MHz) |
|---|---|---|
| | 3 | 917.250 |
| | 4 | 917.750 |
| | 5 | 918.250 |
| | 6 | 918.750 |
| | 7 | 919.250 |
| | 8 | 919.750 |
| | 9 | 920.250 |
| | 10 | 920.750 |
| | 11 | 921.250 |
| | 12 | 921.750 |
| | 13 | 922.250 |
| | 14 | 922.750 |
| | 15 | 923.250 |
| | 16 | 923.750 |
| | 17 | 924.250 |
| | 18 | 924.750 |
| | 19 | 925.250 |
| | 20 | 925.750 |
| | 21 | 926.250 |
| | 22 | 926.750 |
| | 23 | 927.250 |
| 19: Vietnam 920-925 MHz | 1 | 920.250 |
| | 2 | 920.750 |
| | 3 | 921.250 |
| | 4 | 921.750 |
| | 5 | 922.250 |
| | 6 | 922.750 |
| | 7 | 923.250 |
| | 8 | 923.750 |
| | 9 | 924.250 |
| | 10 | 924.750 |
| 20: Israel 915-917 MHz | 1 | 916.250 |
| 21: Philippines 918-920 MHz | 1 | 918.250 |
| | 2 | 918.750 |
| | 3 | 919.250 |
| | 4 | 919.750 |
| 22: Canada Post 902-928 MHz | 1 | 902.750 |
| | 2 | 903.250 |
| | 3 | 905.750 |

| Region | ChannelIndex | Frequency (MHz) |
|---|---|---|
| | 4 | 906.250 |
| | 5 | 906.750 |
| | 6 | 907.250 |
| | 7 | 907.750 |
| | 8 | 908.250 |
| | 9 | 908.750 |
| | 10 | 909.250 |
| | 11 | 909.750 |
| | 12 | 910.250 |
| | 13 | 910.750 |
| | 14 | 911.250 |
| | 15 | 911.750 |
| | 16 | 912.250 |
| | 17 | 912.750 |
| | 18 | 913.250 |
| | 19 | 913.750 |
| | 20 | 914.250 |
| | 21 | 914.750 |
| | 22 | 915.250 |
| | 23 | 917.750 |
| | 24 | 918.250 |
| | 25 | 918.750 |
| | 26 | 919.250 |
| | 27 | 919.750 |
| | 28 | 920.250 |
| | 29 | 920.750 |
| | 30 | 921.250 |
| | 31 | 921.750 |
| | 32 | 922.250 |
| | 33 | 922.750 |
| | 34 | 923.250 |
| | 35 | 923.750 |
| | 36 | 924.250 |
| | 37 | 924.750 |
| | 38 | 925.250 |
| | 39 | 925.750 |
| | 40 | 926.250 |
| | 41 | 926.750 |
| | 42 | 927.250 |

| Region | ChannelIndex | Frequency (MHz) |
|---|---|---|
| 23: Indonesia 923-925 MHz | 42 | 923.250 |
| | 43 | 923.750 |
| | 44 | 924.250 |
| | 45 | 924.750 |
| 24: New Zealand 921.5-928 MHz | 40 | 922.250 |
| | 41 | 922.750 |
| | 42 | 923.250 |
| | 43 | 923.750 |
| | 44 | 924.250 |
| | 45 | 924.750 |
| | 46 | 925.250 |
| | 47 | 925.750 |
| | 48 | 926.250 |
| | 49 | 926.750 |
| | 50 | 927.250 |
| 25: Japan 916.7-920.9 MHz | 1 | 916.8 |
| | 2 | 918.0 |
| | 3 | 919.2 |
| | 4 | 920.4 |
| 26: Latin America 902-928 MHz | 1 | 902.750 |
| | 2 | 903.250 |
| | 3 | 903.750 |
| | 4 | 904.250 |
| | 5 | 904.750 |
| | 6 | 905.250 |
| | 7 | 905.750 |
| | 8 | 906.250 |
| | 9 | 906.750 |
| | 10 | 907.250 |
| | 11 | 907.750 |
| | 12 | 908.250 |
| | 13 | 908.750 |
| | 14 | 909.250 |
| | 15 | 909.750 |
| | 16 | 910.250 |
| | 17 | 910.750 |
| | 18 | 911.250 |
| | 19 | 911.750 |
| | 20 | 912.250 |

| Region | ChannelIndex | Frequency (MHz) |
|---|---|---|
| | 21 | 912.750 |
| | 22 | 913.250 |
| | 23 | 913.750 |
| | 24 | 914.250 |
| | 25 | 914.750 |
| | 26 | 915.250 |
| | 27 | 915.750 |
| | 28 | 916.250 |
| | 29 | 916.750 |
| | 30 | 917.250 |
| | 31 | 917.750 |
| | 32 | 918.250 |
| | 33 | 918.750 |
| | 34 | 919.250 |
| | 35 | 919.750 |
| | 36 | 920.250 |
| | 37 | 920.750 |
| | 38 | 921.250 |
| | 39 | 921.750 |
| | 40 | 922.250 |
| | 41 | 922.750 |
| | 42 | 923.250 |
| | 43 | 923.750 |
| | 44 | 924.250 |
| | 45 | 924.750 |
| | 46 | 925.250 |
| | 47 | 925.750 |
| | 48 | 926.250 |
| | 49 | 926.750 |
| | 50 | 927.250 |
| 27: Peru 916-928 MHz | 1 | 916.250 |
| | 2 | 916.750 |
| | 3 | 917.250 |
| | 4 | 917.750 |
| | 5 | 918.250 |
| | 6 | 918.750 |
| | 7 | 919.250 |
| | 8 | 919.750 |
| | 9 | 920.250 |

| Region | ChannelIndex | Frequency (MHz) |
|---|---|---|
| | 10 | 920.750 |
| | 11 | 921.250 |
| | 12 | 921.750 |
| | 13 | 922.250 |
| | 14 | 922.750 |
| | 15 | 923.250 |
| | 16 | 923.750 |
| | 17 | 924.250 |
| | 18 | 924.750 |
| | 19 | 925.250 |
| | 20 | 925.750 |
| | 21 | 926.250 |
| | 22 | 926.750 |
| | 23 | 927.250 |

# 10 Revision History

| Date | Revision | Comments |
|---|---|---|
| 02/21/2008 | 3.0 | Original document created for Octane 3.0 release |
| 01/12/2009 | 3.2 | Updated for Octane 3.2 release |
| 04/24/2009 | 4.0 | Updated for Octane 4.0 release |
| 08/12/2009 | 4.2 | Updated for Octane 4.2 release |
| 04/07/2010 | 4.4 | Updated for Octane 4.4 release |
| 05/26/2010 | 4.4 rev 2 | Updated for Octane 4.4.1 release |
| 11/1/2010 | 4.6 | Updated for Octane 4.6 release |
| 12/1/10 | 4.6 rev 2 | Updated for Octane 4.6 release |
| 4/25/11 | 4.8 | Updated for Octane 4.8 release. Removed Speedway-specifics. Document going forward covers only Speedway Revolution. |
| 10/27/11 | 4.8.2 | Updated to reflect Gen2 mode 5 addtion for ETSI based regions. |
| 1/23/12 | 4.10 | Added new Japan 916-920MHz region, region 25 and Latin America 902-928MHz region, region 26. |
| 9/30/14 | 5.0 | Added new Peru 916-928MHz region, region 27<br>Updated ImpinjRequestedData enumeration<br>Added ImpinjIntelligentAntennaManagement |
| 12/16/14 | 5.2 | Added xArray specific LLRP content<br>LTK version 10.18.1.240 |

**Notices:**