



Guide

# IMPINJ R700 EMBEDDED TOOLKIT

## QUICK START GUIDE

# TABLE OF CONTENTS

<b>1</b>	<b>Configuring a Development Environment.....</b>	<b>2</b>
1.1	Create a Linux Environment .....	2
1.2	Download the ETK and LTK .....	2
1.3	Extract the R700 ETK .....	2
<b>2</b>	<b>Configuring and Compiling a sample application.....</b>	<b>2</b>
2.1	Extract the LTK .....	2
2.2	Compile the Application .....	2
2.3	Create a CAP description file .....	2
2.4	Create CAP directory .....	3
2.5	Move built application to CAP directory .....	3
2.6	Create a reader config file.....	3
2.7	Create a .upgx CAP file .....	3
<b>3</b>	<b>Deploying and running The Application .....</b>	<b>4</b>
3.1	Deploying the CAP.....	4
3.2	Running the Application .....	4
<b>4</b>	<b>Notices.....</b>	<b>5</b>

# 1 CONFIGURING A DEVELOPMENT ENVIRONMENT

## 1.1 Create a Linux Environment

Using your preferred method (Desktop, Virtual Machine, Vagrant, etc.) create a Linux environment. This guide is written with Ubuntu 19.10.

## 1.2 Download the ETK and LTK

From the support portal download `7.0.0_Octane_Embedded_Development_Tools.tar.gz` and `libltkcpp-linux-10.36.0.0-sdk.tar.gz`. Place them in the same folder.

## 1.3 Extract the R700 ETK

Navigate to the folder where the tarballs are located. Use the following commands to extract the the ETK and copy the references to the expected location for the sample application Makefile.

```
tar xfv 7.0.0_Octane_Embedded_Development_Tools.tar.gz
sudo mkdir -p /opt/armv7l/arm-toolchain
sudo cp -r 7.0.0_Octane_Embedded_Development_Tools/arm-toolchain/
/opt/armv7l/
```

Now your environment is ready to compile code to run on the Impinj R700.

# 2 CONFIGURING AND COMPILING A SAMPLE APPLICATION

For this guide we will use a C++ example that ships with the Impinj Octane LLRP Toolkit to create our application.

## 2.1 Extract the LTK

Run the following commands to extract the LTK.

```
tar xfv ~/libltkcpp-linux-10.36.0.0-sdk.tar.gz
```

## 2.2 Compile the Application

Switch to the `docsample1` directory.

```
cd libltkcpp-linux/docsample1
```

Compile the sample application for the R700 processor architecture.

```
make armv7l
```

## 2.3 Create a CAP description file

Using the text editor of your choice, in the `docsample1` folder create `cap_description.in` and copy the following contents into the file

```
# Upgrade Description File
#
# This file contains the settings used by the upgrade generation tool
# to produce an Impinj firmware upgrade file.
[Description]

# Version is a 4 part number in decimal with each part limited to
# 0-255. It is the version of the upgrade file to be generated.
Version = 0.0.0.1

# Valid Reader Hardware is a 3 part number in decimal representing
# the reader model and major/minor revisions on which the upgrade may be
# loaded. Each field may be replaced by the wildcard '*' or '255',
# which matches any.

Valid Reader Hardware = R700

# Input Directory is the top-level directory of the filesystem to
# create. The files under this directory will be available on the
# reader under /cust after the upgrade is loaded.
Input Directory = ./cap
```

## 2.4 Create CAP directory

Run the following commands to create the cap and sys directories from the docsample1/ directory.

```
mkdir cap
mkdir cap/sys
```

## 2.5 Move built application to CAP directory

```
mv docsample1_armv7l cap/
```

## 2.6 Create a reader config file

To run the application on the reader we need to update the oshell password using a reader config file. Using the text editor of your choice copy the following text into cap/sys/reader.conf

```
[rshell]
password=developer

[SoftwareFeatures]
StartFTP=yes
```

## 2.7 Create a .upgx CAP file

From the /docsample1 folder run the following command.

```
../../../../7.0.0_Octane_Embedded_Development_Tools/cap_gen.sh -d
cap_description.in -o cap.upgx
```

## 3 DEPLOYING AND RUNNING THE APPLICATION

### 3.1 Deploying the CAP

Using a web browser navigate to your reader's WebUI, <http://impinj-AA-BB-CC> where AA-BB-CC are the last three octets of your reader's MAC address. Select the Browse button and navigate to the location of the cap.upgx file. Select the Upgrade button. When the reader says reader for reboot, reboot the reader. After reboot take note of the reader's IP address.

### 3.2 Running the Application

Using putty SSH into the reader. When connected to RShell use the following command to get access to the linux shell.

```
osshell developer
```

Use the following commands to navigate to the /cust folder and execute the application.

```
cd /cust  
./docsample1_armv7l [ip address of reader]
```

The application will print to screen tags that are in the field of view of the reader.

## 4 NOTICES

Copyright © 2020, Impinj, Inc. All rights reserved.

Impinj gives no representation or warranty, express or implied, for accuracy or reliability of information in this document. Impinj reserves the right to change its products and services and this information at any time without notice.

EXCEPT AS PROVIDED IN IMPINJ'S TERMS AND CONDITIONS OF SALE (OR AS OTHERWISE AGREED IN A VALID WRITTEN INDIVIDUAL AGREEMENT WITH IMPINJ), IMPINJ ASSUMES NO LIABILITY WHATSOEVER AND IMPINJ DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATED TO SALE AND/OR USE OF IMPINJ PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT.

NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY PATENT, COPYRIGHT, MASKWORK RIGHT, OR OTHER INTELLECTUAL PROPERTY RIGHT IS GRANTED BY THIS DOCUMENT.

Impinj assumes no liability for applications assistance or customer product design. Customers should provide adequate design and operating safeguards to minimize risks.

Impinj products are not designed, warranted or authorized for use in any product or application where a malfunction may reasonably be expected to cause personal injury or death, or property or environmental damage ("hazardous uses"), including but not limited to military applications; life-support systems; aircraft control, navigation or communication; air-traffic management; or in the design, construction, operation, or maintenance of a nuclear facility. Customers must indemnify Impinj against any damages arising out of the use of Impinj products in any hazardous uses.

Statements concerning Impinj product performance are based on Impinj's internal modeling and test data, actual results may vary.

Impinj, and Impinj products and features are trademarks or registered trademarks of Impinj, Inc. For a complete list of Impinj Trademarks, visit [www.impinj.com/trademarks](http://www.impinj.com/trademarks). All other product or service names may be trademarks of their respective companies.

The products referenced in this document may be covered by one or more U.S. patents. See [www.impinj.com/patents](http://www.impinj.com/patents) for details.