

# ADQUpdater

## User Guide

**Author(s):** Teledyne SP Devices  
**Document ID:** 18-2059  
**Classification:** Public  
**Revision:** F  
**Print date:** 2022-04-08

## Contents

<b>1 Introduction</b>	<b>3</b>
1.1 Firmware Image Files . . . . .	3
<b>2 Installation</b>	<b>4</b>
2.1 Principle of Operation . . . . .	4
<b>3 Updating Firmware</b>	<b>5</b>
3.1 Programming an Image . . . . .	5
3.1.1 Memory Regions . . . . .	6
3.2 Changing the Default Image . . . . .	7
3.2.1 USB . . . . .	8
3.2.2 PCIe, PXIe and MTCA . . . . .	8
3.3 Booting from an Image . . . . .	9
3.4 Updating the Configuration Manager . . . . .	9
3.5 Updating the System Manager . . . . .	10

## Document History

Revision	Date	Section	Description	Author
F	2022-04-08	-	Revised for style and consistency	TSPD
		<a href="#">3.2.2</a>	Add note about host systems which make it impossible to follow the fail-safe switch procedure.	TSPD
E	2021-02-04	-	Add information on ADQ3 series digitizers and the system manager	TSPD
		<a href="#">3.4</a>	Add note that updating the configuration manager's firmware is only supported over USB.	TSPD
D	2019-07-29	<a href="#">3.1.1</a>	Add section on memory regions	TSPD
C	2019-07-09	<a href="#">1.1</a>	Add section on firmware image files	TSPD
B	2019-06-28	-	Update document for ADQ8	TSPD
A	2019-01-09	-	Revision A	TSPD

# 1 Introduction

This document is the user guide of the ADQUpdater application which is used to manage the firmware on ADQ7, ADQ8, and ADQ3 series digitizers by Teledyne SP Devices.

## Note

This document is only valid for the ADQ7, ADQ8 and ADQ3 series digitizer models. For other ADQ digitizers see [1].

## Important

When an incompatible version of the ADQUpdater application (earlier than r46217) interfaces with a configuration manager of version 0.9 or later, no operations will be permitted and no images will be available for listing. This is a locking mechanism to prevent corrupting the memory contents. Updating the ADQUpdater application to a later version will resolve this situation.

## Important

Downgrading the configuration manager from v0.9 is *not* supported and *will* result in corrupted data.

## 1.1 Firmware Image Files

The digitizer firmware is provided as a *firmware image file*. The firmware image file may have two different file extensions: `.mcs` and `.mmcs`. The `.mmcs` (meta-MCS) file is an archive which bundles multiple `.mcs` files together with corresponding metadata. To list the contents of the `.mmcs` file, issue

```
$ adqupdater --show-bundle --file=<Path to image>/<image>.mmcs
```

For `.mmcs` files, ADQUpdater will verify that the firmware image matches the digitizer hardware before proceeding with the upload. The `.mmcs` file is used in the same way as the `.mcs` file for all ADQUpdater commands in this document.

## Note

The `.mmcs` file format is supported by ADQUpdater r45770 or later.

## 2 Installation

ADQUpdater is installed as a part of the SDK installer on Windows systems. On Linux systems, the application is installed as a part of the `adqt001s` package. After a successful installation, the application is located in:

```
<Path to installation directory>/ADQUpdater/
```

on Windows systems. This is normally

```
C:/Program Files/SP Devices/ADQUpdater/
```

On Linux systems, the tool should be available on the system path and be accessible from a terminal.

### 2.1 Principle of Operation

The application is centered around performing one or several (closely related) actions each time it is executed. Which action to perform is determined by the options passed on the command line. A list of the available options for ADQUpdater can be viewed by calling

```
$ adqupdater --help
```

#### **Note**

Double-dashed options accepting an argument allow an equals sign *or* spaces between the option and the argument. For the single-dashed options, only spaces are allowed.

## 3 Updating Firmware

ADQ7 and ADQ8 feature a *configuration manager* which handles the startup procedure at power-on as well as the management of firmware images. On ADQ3 series digitizers, the same component is present but is called the *system manager* to indicate that it takes care of more tasks than just firmware management. Up to four different firmware images can be stored on the digitizer with one of the images specified as the *default image*. This is the firmware loaded when the digitizer is powered on.

### 3.1 Programming an Image

#### Important

Version 0.9 of the configuration manager introduces the concept of *memory regions* to be able to support firmware images of arbitrary size. Refer to Section 3.1.1 for more information.

Given a compatible firmware image (.mcs or .mmcs file), follow the steps detailed below to transfer the image to the digitizer.

1. Power up the digitizer and verify that the power LED lights up.
2. On the host computer, start a command prompt or terminal window with access to the ADQUpdater application.
3. List the available devices with

```
$ adqupdater --list
```

Make sure the digitizer is not being used by another process, otherwise the above command will indicate that the device is busy.

4. Note the device number *N* and list the digitizer's firmware images with

```
$ adqupdater -d N --list-images
```

5. Select an appropriate image number *M* and proceed to erase, update and verify the target image with

```
$ adqupdater -d N -i M -euv --file=<Path to image>/<image>.mcs
```

The above command will attempt to extract the image's metadata (a description and a revision number) from the filename and the result should be sufficient for most use cases. To instead enter this data manually, a short description of the image file can be specified with the option `--imagedescr=X`. The description *X* is limited to 16 characters. Additionally, the contents of the revision field can be modified with the option `--imagerev=X`, where *X* is an unsigned 32-bit number. For example,

```
$ adqupdater -d1 -i2 -euv --file=<Path to image>/<image>.mcs  
--imagedescr ADQ7-1CH-FWDAQ --imagerev 37812
```

will upgrade image 2 on device 1 with the contents of <image>.mcs. The image's description and its revision will be 'ADQ7-1CH-FWDAQ' and '37812', respectively.

### Note

An .mmcs file bundles metadata together with the firmware image data. The the image description and revision information is not derived from the filename if a file of this type is used.

- (OPTIONAL) Verify that the digitizer image list reflects the changes with

```
$ adqupdater -d N --list-images
```

### 3.1.1 Memory Regions

Version 0.9 of the configuration manager introduces the concept of *memory regions* to support firmware images of arbitrary size. Earlier versions only supports firmware images of at most 32 MB. However, this size restriction is not enforced and programming an image larger than 32 MB using an older configuration chain may corrupt the images currently in memory.

The available memory is divided into regions and a firmware image may claim several of these regions, depending on its size. Two firmware images cannot occupy the same region and the regions claimed by an image must constitute a contiguous block of memory. An image is placed starting at the *first* available region which begins a block of contiguous memory of sufficient size. If there is not enough free memory available for the target image, the user will be requested to erase another image. To display the current memory layout, issue

```
$ adqupdater -d N --show-memory-layout
```

Three example layouts are shown below:

Layout 1	Layout 2	Layout 3
-----	-----	-----
Region   Image   Size	Region   Image   Size	Region   Image   Size
-----	-----	-----
0      0    16 MB	0      0    16 MB	0      0    16 MB
1      0    16 MB	1      0    16 MB	1      0    16 MB
2      1    16 MB	2      1    16 MB	2      -    16 MB
3      1    16 MB	3      1    16 MB	3      -    16 MB
4      2    16 MB	4      1    16 MB	4      1    16 MB
5      2    16 MB	5      3    16 MB	5      1    16 MB
6      -    16 MB	6      3    16 MB	6      1    16 MB
7      -    15 MB	7      3    15 MB	7      -    15 MB
-----	-----	-----

### Layout 1

The device is programmed with three firmware images, numbered 0–2. Each image claims two regions and regions 6 and 7 are unclaimed.

### Layout 2

The device is programmed with three firmware images: 0, 1 and 3. Image 0 fits within the first 32 MB while the other two images both require an additional region to support their size.

### Layout 3

The device is programmed with two firmware images: 0 and 1. Image 1 is larger than image 0 and claims three regions of memory. Due to the layout, programming a new firmware image is only allowed if the size does not exceed 32 MB since there is no larger contiguous space available.

## 3.2 Changing the Default Image

The *default image* is loaded automatically when the digitizer is powered on. To change the default image, there are two methods available to the user—which one to use depends on with which interface the digitizer is connected to the host computer.

### USB

The default image may be updated immediately. See Section 3.2.1 for details. The new image is loaded when the device is power cycled. To immediately boot from an image, refer to Section 3.3.

### PCIe, PXIe and MTCA

ADQUpdater will force the user to use a *fail-safe* method to update the default image.

#### Note

This method for PCIe, PXIe and MTCA devices requires support for this functionality across several components of the digitizer. If support is missing, the 'old' way is still available to the user and is identical to the USB case described in Section 3.2.1. Full support requires

- digitizer firmware r41832 or later and
- configuration manager v0.6 or later.

#### Note

This method is currently not supported for ADQ3 series digitizers. Use same commands as for the USB method described in Section 3.2.1 instead.

The process outline is as follows:

1. The new firmware is uploaded to the device.
2. The new firmware image is selected as a *candidate* image. The next time the digitizer is power cycled, this new firmware image is loaded instead of the *default* firmware image.



3. After power cycling the digitizer, the user attempts to interface with the digitizer. In this step there are two possible outcomes:
  - (a) the digitizer is responsive and works as expected. The user may now *confirm* the candidate image, thus making it the new default image.
  - (b) the digitizer is not responsive or does not behave as expected. The user may now power cycle the device once more, triggering a load of the default firmware which brings the device back to a known working state.
4. The update process is complete. Depending on the outcome of state 3, either the previous firmware is loaded every time the device boots up or from now on, the digitizer will load the updated firmware.

See Section 3.2.2 for details. The fail-safe method is also supported over USB but not enforced by the application.

#### **!** Important

Some host systems may power cycle the digitizer several times during their boot up sequence. This makes it impossible to follow the steps outlined above since step 3b is triggered without the user's involvement. The only option is to bypass the fail-safe mechanism and to set a new default image as described in Section 3.2.1. To bypass the fail-safe mechanism, append the option `--forceop` to the listed command.

### 3.2.1 USB

Repeat steps 1–4 in Section 3.1. Choose a valid target image *M* and update the default image with

```
$ adqupdater -d N --set-default-image=M
```

### 3.2.2 PCIe, PXIe and MTCA

1. Repeat steps 1–4 in Section 3.1. Choose a valid target image *M* and begin the process by specifying a *candidate image* with

```
$ adqupdater -d N --set-candidate-image=M
```

This operation corresponds to step 2 in Section 3.2. At this point, the image *M* has been staged as a candidate image and the next time the digitizer is power cycled, this image will be loaded. However, this only happens *once*, meaning that any subsequent power cycle will fall back to loading the default image (which has not yet changed). The selected image will be marked with a 'c' in the Def column in the output from `--list-images`.

2. Power cycle the device by power cycling the host system.
3. Attempt to communicate with the digitizer through ADQUpdater, e.g. by listing the available images.

```
$ adqupdater -d N --list-images
```

This operation corresponds to step 3 in Section 3.2. If the operation produces the expected output, i.e. a table containing information about the available images and not an error message, it is safe to assume the image can be confirmed (step 3a). Otherwise, restart the host system to bring the digitizer back to its original state (step 3b) and do not proceed any further with these instructions.

#### 4. Confirm the candidate image with

```
$ adqupdater -d N --confirm-candidate-image
```

This operation corresponds to step 3a in Section 3.2. This operation is only available if the current firmware image is a confirmable candidate image. Additionally, `--list-images` marks such an image with a '\*' in the `Def` column.

### 3.3 Booting from an Image

If a digitizer is loaded with several images, it is possible to initiate a switch to another image without restarting the device. However, this procedure is only supported when the digitizer is connected to the host computer via USB. Devices connected via any other interface, i.e. PCIe or PXIe must follow the procedure outlined in Section 3.2.2.

#### Important

Booting from another image is only supported for devices connected to the host computer via USB.

To boot a image, issue

```
$ adqupdater -d N -i M -b
```

where `N` is the target device and `M` is the target image. The device and image number can be obtained by following step 3 and 4 in Section 3.1.

The digitizer is immediately reconfigured with the target image. If any application was using the digitizer at the point of reconfiguration, unpredictable errors are likely to occur.

#### Important

Make sure the digitizer is not being used by another process before initiating a boot from another image.

### 3.4 Updating the Configuration Manager

Updating the configuration manager firmware is carried out by following the steps below.

#### Important

This section is only applicable for ADQ7 and ADQ8. For ADQ3 series digitizers, see Section 3.5

### Important

Updating the configuration manager firmware must be carried out over the USB interface. PCIe, PXIe or MTCA devices must be connected to the host computer via an USB cable to the digitizer's USB port.

1. Power up the digitizer and verify that the power LED lights up.
2. On the host computer, start a command prompt or terminal window with access to the ADQUpdater application.
3. List the available devices with

```
$ adqupdater --list
```

Make sure the digitizer is not being used by another process, otherwise the above command will return an empty list.

4. Note the device number *N* and proceed to erase and update the image with

```
$ adqupdater -d N --fpga=2 -eu --file=<Path to image>/<image>.bit
```

## 3.5 Updating the System Manager

Updating the system manager firmware is carried out by following the steps below.

### Important

This section is only applicable for ADQ3 series digitizers. For ADQ7 and ADQ8, see Section 3.4

1. Power up the digitizer
2. On the host computer, start a command prompt or terminal window with access to the ADQUpdater application.
3. List the available devices with

```
$ adqupdater --list
```

Make sure the digitizer is not being used by another process, otherwise the above command will return an empty list.

4. Note the device number *N* and proceed to erase and update the image with

```
$ adqupdater -d N --sysman --file=<Path to image>/<image>.mcs --force
```

## References

[1] Teledyne Signal Processing Devices Sweden AB, *13-1130 ADQUpdater guide*. Technical Manual.

**Worldwide Sales and Technical Support**

[spdevices.com](https://spdevices.com)

**Teledyne SP Devices Corporate Headquarters**

Teknikringen 8D

SE-583 30 Linköping

Sweden

Phone: +46 (0)13 645 0600

Fax: +46 (0)13 991 3044

Email: [spd\\_info@teledyne.com](mailto:spd_info@teledyne.com)