

[Software Development Kit](#) > [nRF5 SDK](#) > [nRF5 SDK v11.0.0-2.alpha](#) > [Examples](#) > [DFU bootloader examples](#) > [BLE & HCI/UART Bootloader/DFU](#) > [Creating a DFU bootloader](#)

nRF5 SDK v11.0.0-2.alpha

## Safety-checking the image

*This information applies to the following SoftDevices: **S130, S132***

To make sure that only compatible applications are installed on a device, the DFU procedure includes a mechanism to safety-check the transferred firmware image.

When updating the application on the nRF5 IC, the image that is transferred must be accompanied by an init packet that contains information about the image. The tool that you use to perform the DFU must [send this packet](#) before transferring the actual image. The DFU processing in the bootloader must [check the information in the init packet](#) to ensure that the transferred image is valid and to accept only compatible applications.

The init packet contains the following information that is used for safety checks (see [dfu\\_init\\_packet\\_t](#)):

- Device type: A 2-byte value specified by the developer that identifies the device type, for example Heart Rate Belt.
- Device revision: A 2-byte value that can be used to restrict the update to be accepted only on devices with a defined revision number.
- Application version: A 4-byte value identifying the version of the application that is being transferred. This value can be used to allow only software upgrades and prevent downgrades. No example code is provided for this feature.
- Supported SoftDevices: A list of 2-byte values identifying the SoftDevices that are compatible with the application, for example, S132 v2.0.0.
- Checksum: A 2-byte CRC-16-CCITT for the image to transfer.

Device Type	Device Revision
Application Version	
SoftDevice array length (n)	SoftDevice - 1
SoftDevice - 2	.....
SoftDevice - n	CRC - 16 - CCITT
2 bytes	2 bytes

DFU init packet

## Sending the init packet

How to send the init packet depends on the procedure that you use to perform the DFU.

If you use Master Control Panel or a Python script to perform the update, you must provide a zip file that contains the image and the init packet.

If you use Nordic Semiconductor's Android or iOS app to perform the update, the required file format depends on the version of the app. New versions support zip files that contain the firmware image and the init packet. Older versions support BIN or HEX files and will prompt you to select an init packet to send.

## Checking the init packet

The nRF5 SDK provides a template, `dfu_init_template.c`, to perform safety checks of the init packet. The template is located in the `<BaseFolder>\bootloader_dfu` folder. If you are using Keil packs, the default `<BaseFolder>` is

`C:\Keil\ARM\Pack\NordicSemiconductor\nRF_Libraries\<version>`. If you are using the repository distribution variant of the SDK, `<BaseFolder>` is `<InstallFolder>\components\libraries`.

`dfu_init_template.c`, which is also used in the DFU bootloader example projects, can be used as a starting point to develop procedures that increase the safety of the DFU. The current implementation includes checks for [Device type and revision](#), [Supported SoftDevices](#), and the checksum, but not for the [Application version](#).

## Device type and revision

The device type and revision are stored in the user-reserved area of UICR (0x10001080) on the nRF5 IC. If this location is used for other purposes, update the offset [UICR\\_CUSTOMER\\_DEVICE\\_INFO\\_OFFSET](#) in `dfu_init.h` to match a free location in UICR.

```
#define UICR_CUSTOMER_DEVICE_INFO_OFFSET    0x0
/**< Device info offset inside the customer UICR reserved area. Customers
may change this value to place the device information in a user-preferred
location. */
```

The values stored at this location are compared to the values from the init packet. If they match, the image is accepted. Otherwise, the image is rejected. To accept all device types and revisions and to disable the check, make sure not to set the UICR value to a specific value, but keep the default value 0xFFFF.

## Application version

`dfu_init_template.c` does not check the application version. However, you should implement an application version check if required.

If you add an application version check, every application must be compiled with a version ID. This version ID can be placed at a predefined location in the application image, for example at the application start address + 0x0100, similar to the principle used by Nordic Semiconductor's SoftDevices.

See the following code snippet from `dfu_init_template.c`, which illustrates where to extend the DFU Init packet handling with an application version safety check:

```
// To support application versioning, this check should be updated.  
// This template allows for any application to be installed. However,  
// customers can place a revision number at the bottom of the application  
// to be verified by the bootloader. This can be done at a location  
// relative to the application, for example the application start  
// address + 0x0100.
```

## Supported SoftDevices

Applications that are compiled for the nRF5 IC target a specific SoftDevice, for example S132 v2.0.0. Some applications might work with multiple SoftDevice versions if the API is backward compatible. For example, an application that is compiled for S132 SoftDevice v1.0.0 can also run on S132 SoftDevice v2.0.0.

Provide a list of supported SoftDevices for the application that is to be installed in the DFU init packet. The DFU procedure in the bootloader checks the list that is provided in the init packet against the currently installed SoftDevice on the IC and continues the update procedure only if a matching SoftDevice is installed.

Use a value of 0xFFFF in the init packet if the application should be installed regardless of the SoftDevice that is present. This feature can be helpful during development, but you should not use it in a product.

See the following table for the FWID values of current SoftDevices:

SoftDevice	FWID
S130 v2.0.0-7.alpha	0x0078
S132 v2.0.0-7.alpha	0x0079
Development/any	0xFFFF

---

This document was last updated on Fri Dec 18 2015.

Please send us your [feedback](#) about the documentation! For technical questions, visit the [Nordic Developer Zone](#) [<https://devzone.nordicsemi.com/questions/>](https://devzone.nordicsemi.com/questions/).