



µTasker Document

**i.MX RT Serial Loader Memory Layout**



## Table of Contents

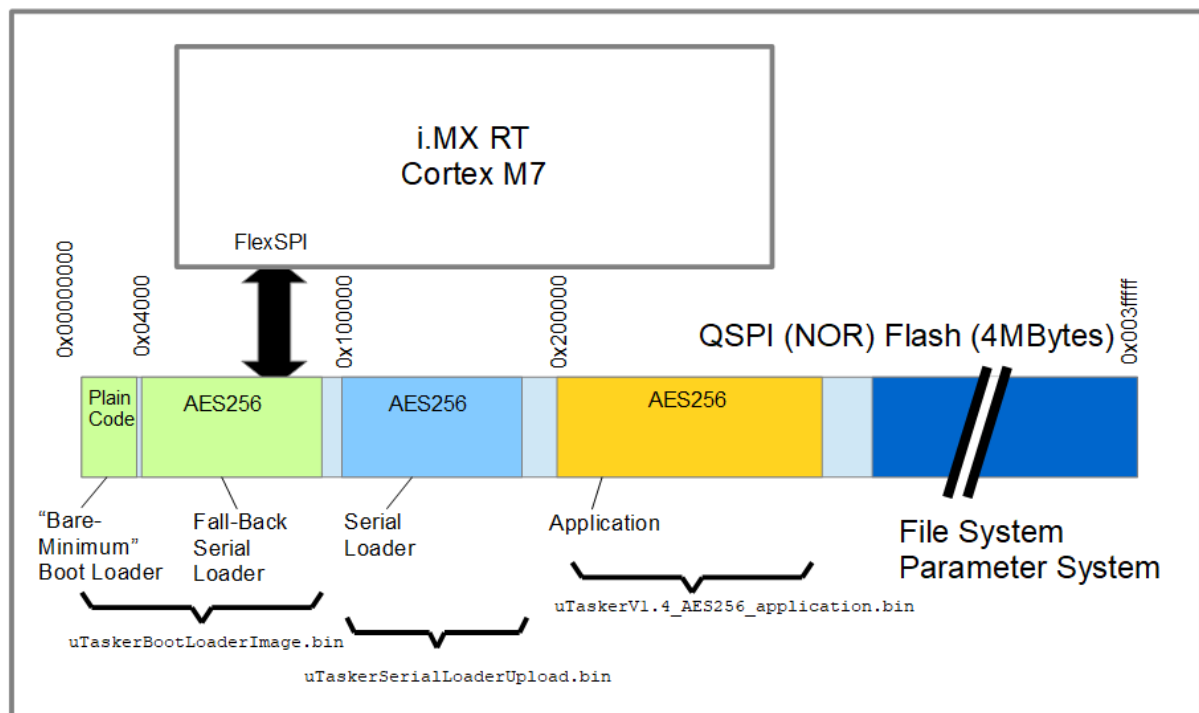
|                                  |   |
|----------------------------------|---|
| 1. Introduction.....             | 3 |
| 2. Default Memory Layout.....    | 3 |
| 3. Customised Memory Layout..... | 4 |
| 4. Conclusion.....               | 6 |

## 1. Introduction

The  $\mu$ Tasker Loader Concept uses a default memory layout in QSPI flash, which is described in this document.

Details relevant to changing the default layout are then given which allows adjusting this in case large serial loaders are required – *examples of larger size serial loaders are those configured with a large amount of features or those including LCD indications of their operations which can require images to be integrated that take up more space than envisaged for typical default usage.*

## 2. Default Memory Layout



This diagram shows the standard layout for typical QSPI Flash memory types that have a 64k (or smaller) erase granularity.

The "Bare-Minimum" Boot Loader occupies the first 16k of the QSPI flash and is followed by the "Fall-back" Serial Loader that can occupy up to 48k of QSPI flash.

The Serial Loader follows, at `0x100000`, and can occupy up to 64k of QSPI flash.

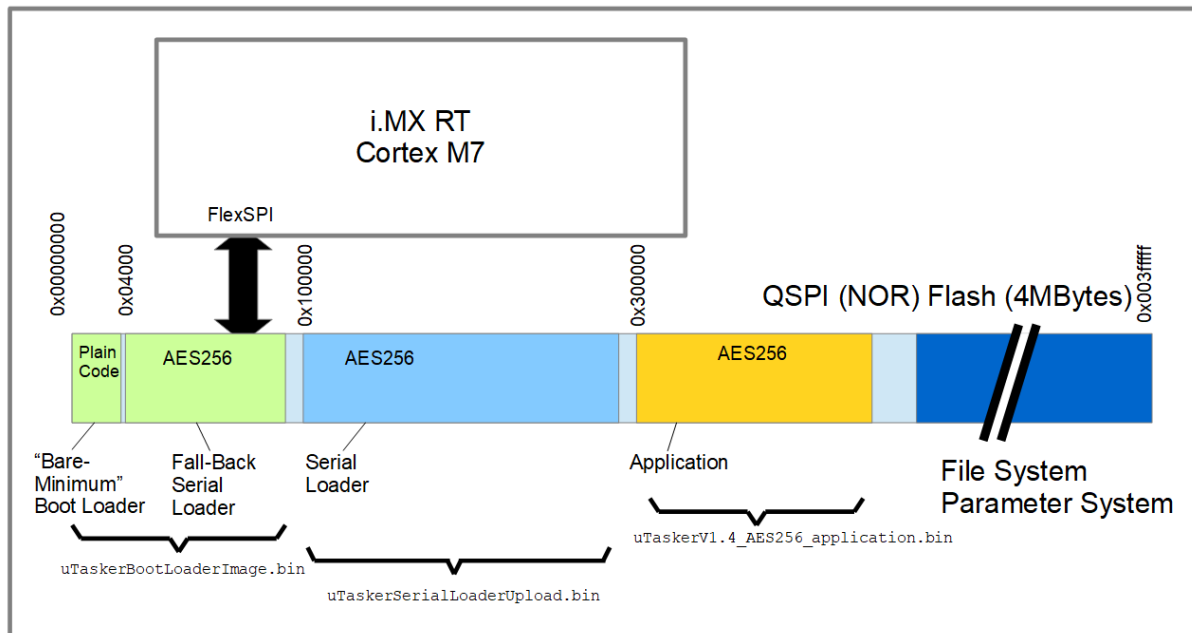
The Application and other consumers of QSPI Flash (like file systems) occupy the remaining QSPI flash, beginning at `0x200000` (after 128k of area reserved for the boot loader concept).

*Execute-in-Place (XiP) application code starts at `0x20400` in this layout, whereby the `0x400` offset from the start of the application area ensures full compatibility with libraries and interrupt vectors located in QSPI.*

### 3. Customised Memory Layout

Assuming that the Serial Loader size surpasses the default 64k it becomes necessary to modify the default layout to make adequate space for it. Although a doubling may not be absolutely necessary (what is possible depends on the QSPI flash type used and its erase granularity) an increase to 128k is used as test case. It is also assumed that the „Fall-back“ serial loader doesn't need additional memory.

The new layout thus becomes:



where it become obvious that the change has an impact on the start of the application (originally at 0x20000 it moves to 0x30000). The remaining size of the SPI flash reduces of course by 64k, *which may or may not have consequences on the application itself.*

1.

Due to this change it is necessary to rebuild the „Bare-Minimum“ Boot Loader with the new location of the application, which can be performed by adding (to `config.h`)

```
#define SERIAL_LOADER_FLASH_AREA_SIZE (128 * 1024)
```

which causes the default size (64k in the case of standard QSPI flash) to be updated with this one and the new application location adjusted accordingly.

2.

The „Fall-back“ Serial Loader needs to be built with the same define added to its `config.h` so that it is aware of the size of the serial loader that it can update.

3.

The Serial Loader needs to be built with the same define (added in `config.h` in step 2) so that it is aware of the start of the application that follows it.

4. *(only XiP)*

If the application operate in XiP mode (directly in QSPI flash) it needs to be linked to its new address `0x30400` instead of `0x20400`.

5. *(only encrypted XiP)*

If the application is to use on-the-fly decryption in XiP it also needs to be appropriately encrypted using

```
uTaskerConvert.exe application.bin aes128_coded_XiP.bin E=128B-60030400 $"aes256
secret key" $"initial vector"
```

instead of the original

```
uTaskerConvert.exe application.bin aes128_coded_XiP.bin E=128B-60020400 $"aes256
secret key" $"initial vector"
```

Note that E=128B used for on-the-fly decryption based on BEE and E=128 based on OTFAD (such as used by the i.MX RT 1011).

6.

Wherever the application finally runs (XiP, RAM or SD-RAM) its location in the image has changed so the following modification is needed when combining the loaders and application into a single, complete image:

```
uTaskerCombine
"../../uTaskerSerialBoot/GNU_iMX/Objects/uTaskerBootComplete_iMX_RT1020.bin"
"Objects/uTaskerV1.4_AES256_iMX_RT1020 .bin" 0x30000[0x100-UPLOAD_DISK]
Objects/uTaskerCompleteImage_iMX_RT1020.bin
Objects/uTaskerCompleteImage_iMX_RT1020.hex
```

instead of

```
"../../uTaskerSerialBoot/GNU_iMX/Objects/uTaskerBootComplete_iMX_RT1020.bin"
"Objects/uTaskerV1.4_AES256_iMX_RT1020.bin" 0x20000[0x100-UPLOAD_DISK]
Objects/uTaskerCompleteImage_iMX_RT1020.bin
Objects/uTaskerCompleteImage_iMX_RT1020.hex
```

## 4. Conclusion

This document has illustrated the memory layout in QSPI flash as used by the  $\mu$ Tasker Loader concept, first showing its „default“ layout for QSPI flash types with smaller or equal to 64k erase granularity. The default layout is suitable without modifications form the majority of projects that don't require a Serial Loader configuration of more than 64k code size.

The modifications to the configuration of the loaders, as well as linking and converting the application for larger Serial Loaders has been shown in order to give a simple step-by-step modification guide for use in such circumstances.