

MOST

Media Oriented Systems Transport

Multimedia and Control
Networking Technology

MOST FBlock Cookbook

Rev 3.1

07/2017

MOSTCO CONFIDENTIAL

See page 3 for the terms of disclosure



Legal Notice

COPYRIGHT

© Copyright 1999 - 2017 MOST Cooperation. All rights reserved.

LICENSE DISCLAIMER

Nothing on any MOST Cooperation Web Site, or in any MOST Cooperation document, shall be construed as conferring any license under any of the MOST Cooperation or its members or any third party's intellectual property rights, whether by estoppel, implication, or otherwise.

CONTENT AND LIABILITY DISCLAIMER

MOST Cooperation or its members shall not be responsible for any errors or omissions contained at any MOST Cooperation Web Site, or in any MOST Cooperation document, and reserves the right to make changes without notice. Accordingly, all MOST Cooperation and third party information is provided "AS IS". In addition, MOST Cooperation or its members are not responsible for the content of any other Web Site linked to any MOST Cooperation Web Site. Links are provided as Internet navigation tools only.

MOST COOPERATION AND ITS MEMBERS DISCLAIM ALL WARRANTIES WITH REGARD TO THE INFORMATION (INCLUDING ANY SOFTWARE) PROVIDED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. Some jurisdictions do not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

In no event shall MOST Cooperation or its members be liable for any damages whatsoever, and in particular MOST Cooperation or its members shall not be liable for special, indirect, consequential, or incidental damages, or damages for lost profits, loss of revenue, or loss of use, arising out of or related to any MOST Cooperation Web Site, any MOST Cooperation document, or the information contained in it, whether such damages arise in contract, negligence, tort, under statute, in equity, at law or otherwise.

FEEDBACK INFORMATION

Any information provided to MOST Cooperation in connection with any MOST Cooperation Web Site, or any MOST Cooperation document, shall be provided by the submitter and received by MOST Cooperation on a non-confidential basis. MOST Cooperation shall be free to use such information on an unrestricted basis.

TRADEMARKS

MOST Cooperation and its members prohibit the unauthorized use of any of their trademarks. MOST Cooperation specifically prohibits the use of the MOST Cooperation LOGO unless the use is approved by the Steering Committee of MOST Cooperation.

SUPPORT AND FURTHER INFORMATION

For more information on the MOST technology, please contact:

MOST Cooperation
Administration
Emmy-Noether-Str. 14
76131 Karlsruhe
Germany
Tel: (+49) (0) 721 966 50 00
E-mail: contact@mostcooperation.com
Web: www.mostcooperation.com



This Specification is Confidential Information of the MOST Cooperation. It may only be disclosed to member companies. Member companies wishing to discuss these Specifications with suppliers or other third parties must ensure that a commercially standard form of non-disclosure agreement has been previously executed by the party receiving such Specifications. Use of these Specifications may only be for purposes for which they are intended by the MOST Cooperation. Unauthorized use or disclosure is a violation of law.

© Copyright 1999 - 2017 MOST Cooperation
All rights reserved

MOST is a registered trademark

Contents

BIBLIOGRAPHY	5
DOCUMENT HISTORY	6
1 INTRODUCTION	7
2 DEFINITIONS	8
2.1 Sections of FktIDs	8
2.2 FBlock	9
2.3 Function Catalog	9
2.3.1 Example of a Function Catalog	10
3 GENERAL STRUCTURE OF FBLOCKS	11
3.1 Static Interface	11
3.1.1 Version of FBlock	11
4 HOW TO BUILD A NEW FBLOCK	12
4.1 Definition of new FBLOCKS, based on the GeneralFBlock	12
4.2 Definition of new FBlock "from scratch"	15
4.2.1 Version upgrade of FBLOCKS	15
5 HOW TO COMPILE A FUNCTION CATALOG (SYSTEM INTEGRATOR)	16
LIST OF FIGURES	17
LIST OF TABLES	18

Bibliography

All documents which this MOST document have references to are listed here with the actual revision this document is referring to.

Number	Document	Revision
1	MOST Specification	3.1

Document History

Changes in MOST FBlock Cookbook Rev. 1.0 to Rev. 3.1

Change Ref.	Section	Changes
3V1_001	All	<ul style="list-style-type: none"> – Removed references to MOST Specification revisions 2.5 and 3.0. – Removed mention of mandatory MSCs for FBlock specifications.
3V1_002	2	Completely revised and updated to match FCat Schema 3.1
3V1_003	2.1	Updated FktID ranges; removed "Unique" range.
3V1_004	3.1.1	Either function Version or FBlockInfo has to be used.
3V1_005	4.1	<ul style="list-style-type: none"> – Consolidated description. – Modified figure so that no FBlock templates are shown. – Removed statement that along with proprietary extended functions also the original MOSTCO function has to be implemented. – Removed statement that an expanded function must not exceed the MOSTCO defined range in the return values. – Removed reference to system specific Enum range A0₁₆. – Using Set when SetGet is present is now a recommendation.
3V1_006	4.2	Corrected and updated FBlockID/FktID responsibilities table.
3V1_007	4.2.1	Clarified that a deleted function should remain unused within the current major version.
3V1_008	5	<ul style="list-style-type: none"> – Removed misleading statements, consolidated. – Supplier specific functions do not have to be included in the System Integrator's function catalog.

Changes in MOST FBlock Cookbook 1V0-00

Change Ref.	Section	Changes
-	-	Initial Revision

1 Introduction

This document can be used for developing MOST function catalogs, FBlocks, or single functions, based on MOST Specification Rev 3.1.

It also can be used as guideline for expanding existing FBlocks, definition of new FBlocks, and for composing function catalogs on the basis of existing FBlocks.

2 Definitions

This chapter provides an overview of some elementary terms, which will be used in conjunction with FBlocks and function catalogs.

The MOST Cooperation provides a “MOST Cooperation FBlock Library”, which is the basis for the creation of function catalogs. The FBlock library contains FBlocks and FBlock templates. FBlock templates do not have an FBlockID.

2.1 Sections of FktIDs

The FktID address range is divided into several sections. Every function of an FBlock is assigned to a section, depending on its FktID:

Section	Address Range	Definition
Coordination	000 ₁₆ - 1FF ₁₆	Functions for administrative purposes in an FBlock.
Application	200 ₁₆ - BFF ₁₆	Functions in the Application range represent the main functionality of an FBlock. Depending on the FBlock range, they are defined either by the MOST Cooperation, the System Integrator, or the Supplier.
Proprietary / System specific	C00 ₁₆ - EFF ₁₆	Functions that can be used by any System Integrator (e.g., car manufacturer). They are specific to a system and are coordinated by the System Integrator between the suppliers developing devices for this system.
Proprietary / Supplier specific	F00 ₁₆ - FFF ₁₆	Functions, which can be used by suppliers for any proprietary purpose.

Table 1: Sections of FktIDs (Overview)

Functions are classified as being mandatory, conditional, or optional.

2.2 FBlock

A MOST FBlock is a collection of standardized functions for a dedicated component (e.g., tuner), provided by the MOST Cooperation.

Additionally, the System Integrator or a supplier may specify proprietary FBlocks.

For the MOST FBlock Library, each FBlock consists of these files:

XML file	The XML file is the main description file for any FBlock. It can be edited with the "MOST Editor". It includes all necessary information about the static portion of the FBlock interface (including the change list). The XML file is the master document of the FBlock.
FCat XML Schema file	The FCat XML Schema file contains the structural definition of the function catalog XML file format. Function catalog XML files are required to validate against the FCat XML Schema file.
HTML file	The HTML file is generated from the XML file, e.g., by the MOST Editor. It is a human readable representation of the static FBlock definition.
CSS file	The CSS file contains the definition of styles that the HTML file relies on for formatting.
PDF file	At least one PDF file has to be distributed with the FBlock. It contains the history of the FBlock (change list) and the static description of the FBlock. This file is based on the HTML file.

If new FBlocks are defined, the files listed above are provided.

Refer to the *MOST FCat Schema Cookbook* for detailed information about the FCat XML Schema file and additional rules for creating FBlock specifications in XML.

2.3 Function Catalog

A function catalog is a collection of FBlocks or independent functions. A MOST function catalog is a collection of MOST FBlocks or independent functions that are defined by the MOST Cooperation.

Apart from potential additions by the System Integrator or supplier, function catalogs are based on the same set of files as FBlocks (see 2.2).

A function catalog may contain information provided by the System Integrator that is not covered by the MOST FBlock specification (e.g., which function uses notification, communication via Control Channel or Packet Data Channel) or restriction of the FBlock specification for the specific system (e.g., reduced range of values). This results in additional effort for the System Integrator.

2.3.1 Example of a Function Catalog

```
<FunctionCatalog xmlns="http://www.mostcooperation.com"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.mostcooperation.com mostfcatalog_3_1_0.xsd">
  <CatalogVersion Release="3.1.0" Date="2015-11-19+01:00" Time="00:00:00.000+01:00"
SchemaVersion="3.1.0">
    <Author>WG DA</Author>
    <Company>MOST Cooperation</Company>
    <!--Generated by MOST Editor 3.4.0.8-->
  </CatalogVersion>
  <SharedParameters>
    <SharedParameter Name="FBlockID_0">
      ...
    </SharedParameter>
    ...
  </SharedParameters>
  <FBlock Id="2" Name="NetworkMaster">
    <FBlockVersion Release="3.1.0" Date="2015-11-19+01:00" Time="00:00:00.000+01:00"
Access="public">
      <Author>WG DA</Author>
      <Company>MOST Cooperation</Company>
    </FBlockVersion>
    <Function Name="Configuration" Id="2560" Virtual="false" Wellknown="false"
Occurrence="Mandatory">
      ...
    </Function>
    <Function Name="CentralRegistry" Id="2561" Virtual="false" Wellknown="false"
Occurrence="Mandatory">
      ...
    </Function>
  </FBlock>
</FunctionCatalog>
```

New FBlocks can be added as follows:

```
<!--FBlock: X-->
<FBlock Id="193" Name="X">
  <Function> ... </Function>
  <Function> ... </Function>
  <Function> ... </Function>
</FBlock>
<!--FBlock: Y-->
<FBlock Id="194" Name="Y">
  <Function> ... </Function>
  <Function> ... </Function>
  <Function> ... </Function>
</FBlock>
```

3 General structure of FBlocks

Any FBlock is specified by an XML file that describes the static interface.

3.1 Static Interface

The static interface specifies all functions of the FBlock and their format, according to MOST Specification.

To ensure interoperability, all mandatory functions have to be implemented.

3.1.1 Version of FBlock

Any FBlock has to implement either function `Version` (FktID 010₁₆) that contains the version of the FBlock or function `FBlockInfo` (FktID 011₁₆) that contains the FBlock name, the name of the instance, the corresponding MOST Specification version, and the version of the FBlock itself.

With every change, the FBlock gets a new version. Versioning of an individual FBlock is done in accordance with MOST Cooperation organizational procedures:

- The first number equals not backward compatible changes.
- The second number equals enhanced functionality.
- The third number equals typographical changes as well as bug fixes.

4 How to build a new FBlock

4.1 Definition of new FBlocks, based on the GeneralFBlock

The GeneralFBlock template is a collection of functions that are defined by the MOST Cooperation for use in new FBlocks.

The following rules apply:

- New FBlocks are based on the GeneralFBlock template.
- All relevant functions are referenced in the new FBlock.
- It is not allowed to add new *OPTypes* to or delete existing *OPTypes* from functions of FBlocks of the MOST Cooperation.
- When using notification, if *SetGet* is specified, *Set* should be specified as well.

Any implemented function that is based on an original FBlock of the *MOST Cooperation FBlock Library* has to be operative without usage of proprietary functions. This ensures interoperability of all implemented functions, defined by the MOST Cooperation.

Any FBlock specification has to contain

- references to the relevant MOST Specification and the GeneralFBlock,
- static description of all specified functions.

Additional guidelines:

- New functions should be mapped to the appropriate function class.
- Use *Methods* to trigger a process and use *Properties* to query data or obtain updates through notification.

How to compile new FBlocks using the GeneralFBlock template

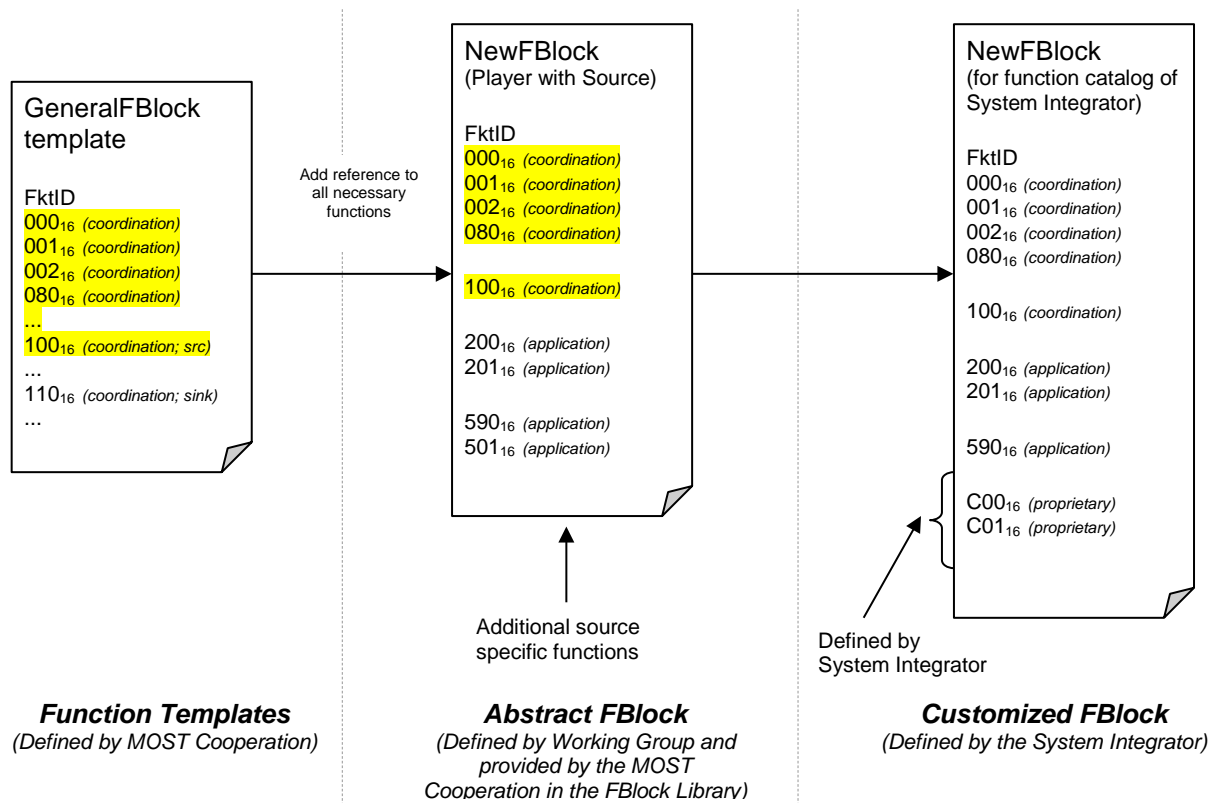


Figure 1: Compiling a new FBlock

During specification of a new FBlock, the following aspects have to be considered:

- Always use the latest version of the GeneralFBlock.
- Add references to the used GeneralFBlock and thereby, implicitly, a reference to the corresponding MOST Specification.
- To increase harmonization, the System Integrator should inform the responsible WG about proprietary functions for potential consideration in future versions of the MOST Cooperation FBlock Library.

In some situations, the functions defined by FBlocks of the MOST Cooperation are not sufficient for System Integrators to implement all necessary features and mechanisms. In that case, the System Integrator is able to introduce new functions, new parameters or new Enum values by consideration of the following:

- If possible, proprietary extensions should be avoided. Before specification of proprietary extension, it has to be checked whether the desired mechanism could be achieved with usage of functions specified by the MOST Cooperation.
- Wherever applicable, introduction of a new FktID within the proprietary FktID range should be preferred instead of extension of an existing FktID that was specified by the MOST Cooperation. The original function should be copied to proprietary FktID range and the extensions could be defined there.
- When introducing, for example, new Enum values, a gap between existing values and new values should be reserved for future extensions by the MOST Cooperation.
- If Enums are extended, a new function has to be defined. This function has to be located in proprietary FktID address range.
- Function and parameter names
 - shall contain no characters other than letters, numbers, and underscores
 - shall not start with a number
 - should not exceed a certain length (e.g., 26 characters)

Design rules for functions that support notification

- It is recommended that an event is designed in a way that it contains all information about the changes.
- For small payload, it is not recommended to signal just the occurrence of an event, which requires a new request by the controller to retrieve additional information about the details. This would lead to inconsistencies.

4.2 Definition of new FBlock “from scratch”

It has to be considered that supplier specific FBlocks shall not be reported via FBlockIDs.Status.

The following table regulates who is authorized to specify certain FBlockID/FktID combinations.

FBlockID	FktID			
	000 ₁₆ ...1FF ₁₆	200 ₁₆ ...BFF ₁₆	C00 ₁₆ ...EFF ₁₆	F00 ₁₆ ...FFF ₁₆
	(Coordination)	(Application)	(System Integrator Specific)	(Supplier Specific)
00 ₁₆ ...0F ₁₆ (MOST Cooperation)	MOST Cooperation	MOST Cooperation	System Integrator	System Integrator
10 ₁₆ ...9F ₁₆ (MOST Cooperation)	MOST Cooperation	System Integrator	System Integrator	Supplier
A0 ₁₆ ...C7 ₁₆ , C9 ₁₆ ...EF ₁₆ (System Specific)	MOST Cooperation	System Integrator	System Integrator	Supplier
F0 ₁₆ ...FE ₁₆ (Supplier Specific)	MOST Cooperation	Supplier	Supplier	Supplier
FF ₁₆	Reserved.			
C8 ₁₆	Reserved for compliance testing.			

Table 2: Responsibilities for FBlockID and FktID ranges

4.2.1 Version upgrade of FBlocks

Upgrading the version of an FBlock, the FktIDs **should** not be changed. Any function that exists in both versions should have the same FktID. If the FktID is changed, the main version of the FBlock has to be changed. This helps avoiding version conflicts, for example, if two versions of an FBlock exists within one MOST system.

If a function is deleted during a version upgrade, the FktID should not be used for a new function. It should remain unused within the current major version.

5 How to compile a Function Catalog (System Integrator)

This chapter describes all necessary steps for a System Integrator to create a function catalog.

The function catalog is a collection of all relevant functions of a MOST system, defined by the System Integrator. Any MOST devices, dedicated for this System Integrator implement a subset of functions from the defined function catalog. The function catalog may contain functions that are not used in the MOST System.

The function catalog provides an overview of all utilized FBlocks and functions of a particular MOST system. It does not provide information about which FBlock and function has to be implemented into each single MOST device of the system.

All FBlocks within one function catalog should be based on the same GeneralFBlock and on the same MOST Specification.

To compile the function catalog, the System Integrator has to copy all used FBlocks with all used functions into the function catalog XML file.

Supplier specific proprietary functions (FktIDs $F00_{16}$ - FFF_{16}) may be included into the function catalog, too.

When generating the function catalog, all references to FBlocks of MOST Cooperation are deleted. However, the version of the MOST Specification the function catalog is based on has to be stored in the function catalog.

The System Integrator may add additional (system specific) information to the function catalog, not covered by the MOST FBlock specification (e.g., which function uses notification, communication via control channel or synchronous channel) or restrict presets of the MOST FBlock specification for the specific system (e.g., reduced range of values).

List of Figures

Figure 1: Compiling a new FBlock.....	13
---------------------------------------	----

List of Tables

Table 1: Sections of FktIDs (Overview)	8
Table 2: Responsibilities for FBlockID and FktID ranges	15

Notes: