

MOST

Media Oriented Systems Transport

Multimedia and Control
Networking Technology

MOST FunctionBlock SDARS

Rev 2.4

06/2004



Legal Notice

COPYRIGHT

© Copyright 1999 - 2004 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
P. O. Box 4327
D-76028 Karlsruhe
Germany

Tel: (+49) (0) 721 966 50 00

Fax: (+49) (0) 721 966 50 01

E-mail: contact@mostcooperation.com

Web: www.mostcooperation.com



© Copyright 1999 - 2004 MOST Cooperation
All rights reserved

MOST is a registered trademark

| | | |
|----------|------------------------------------------------------|-----------|
| 1 | INTRODUCTION | 8 |
| 2 | FUNCTIONBLOCK DEFINITION | 8 |
| 2.1 | Definitions | 8 |
| 2.2 | Introduction | 9 |
| 2.3 | SatRadio (FBlockID = 0x44) | 9 |
| 2.4 | Coordination | 10 |
| 2.4.1 | FktIDs (FktID = 0x000) | 10 |
| 2.4.2 | Notification (FktID = 0x001) | 10 |
| 2.4.3 | NotificationCheck (FktID = 0x002) | 10 |
| 2.4.4 | CreateArrayWindow (FktID = 0x090) | 10 |
| 2.4.5 | DestroyArrayWindow (FktID = 0x091) | 10 |
| 2.4.6 | MoveArrayWindow (FktID = 0x092) | 10 |
| 2.4.7 | SearchArrayWindow (FktID=0x093) | 10 |
| 2.4.8 | LongArrayInfo (FktID=0x094) | 10 |
| 2.4.9 | SourceInfo (FktID = 0x100) | 10 |
| 2.4.10 | Allocate (FktID = 0x101) | 10 |
| 2.4.11 | DeAllocate (FktID = 0x102) | 10 |
| 2.4.12 | SourceConnect (0x105) | 10 |
| 2.4.13 | SourceDisConnect (0x106) | 10 |
| 2.4.14 | SourceRouting (0x107) | 11 |
| 2.4.15 | SourceActivity (FktID = 0x103) | 11 |
| 2.4.16 | SourceName (FktID = 0x104) | 11 |
| 2.4.17 | SyncDataInfo (FktID = 0x116) | 11 |
| 2.5 | Mandatory Functions | 12 |
| 2.5.1 | StationSelect (FktID = 0x210) | 12 |
| 2.5.2 | StationTune (FktID = 0x211) | 13 |
| 2.5.3 | CurrentStationInfo (FktID = 0x212) | 14 |
| | CurrentCategoryInfo (FktID = 213) | 15 |
| 2.5.4 | RadioText (FktID = 0x214) | 16 |
| 2.5.5 | ServiceProvider (FktID = 0x215) | 18 |
| 2.5.6 | ElectronicCode (FktID = 0x216) | 19 |
| 2.5.7 | ServiceStatus (FktID = 0x217) | 20 |
| 2.6 | Optional Functions | 21 |
| 2.6.1 | Presets (FktID = 0x410) | 21 |
| 2.6.2 | TuningStatus (FktID = 0x411) | 23 |
| 2.6.3 | StationListInfo (FktID = 0x412) | 24 |
| 2.6.4 | CategorySelect (FktID = 0x413) | 26 |
| 2.6.5 | CategoryListInfo (FktID = 0x414) | 27 |
| 2.6.6 | StationListforCategory (FktID = 0x415) | 28 |
| 3 | FUNCTIONBLOCK DYNAMIC SPECIFICATION | 29 |
| 3.1 | Startup | 29 |
| 3.1.1 | Notification | 29 |
| 3.2 | Activating the Satellite Radio application | 29 |
| 3.3 | Service Signal Acquisition | 30 |
| 3.4 | Station Selection | 33 |
| 3.4.1 | Direct Access to the Station List | 33 |
| 3.4.2 | Selecting the Station List using Array Windows | 34 |
| 3.4.3 | Select Station from Station List | 36 |
| 3.4.4 | Select Station by Category | 38 |
| 3.4.5 | Scanning Through Stations | 42 |
| 3.4.6 | Station Preset Menu | 44 |
| 3.5 | MFL | 46 |

Bibliography MOST Function Catalog

This is a list of released FunctionBlocks at the release time of this specification. FBlocks released later are not reflected in this list.

| FBlockID | FunctionBlock |
|----------|----------------------|
| 0x00 | GeneralFBlock |
| 0x00 | GeneralPlayer |
| 0x01 | NetBlock |
| 0x02 | NetworkMaster |
| 0x03 | ConnectionMaster |
| 0x06 | Diagnosis |
| 0x0F | Enhanced Testability |
| 0x22 | AudioAmplifier |
| 0x24 | AuxIn |
| 0x26 | MicrophoneInput |
| 0x30 | AudioTapePlayer |
| 0x31 | AudioDiskPlayer |
| 0x34 | DVDVideoPlayer |
| 0x40 | AmFmTuner |
| 0x41 | TMCTuner |
| 0x42 | TVTuner |
| 0x43 | DABTuner |
| 0x44 | SDARS |
| 0x50 | Telephone |
| 0x51 | GeneralPhoneBook |
| 0x60 | GraphicDisplay |
| 0xFF | UniqueFunctions |

FBlock (0x44) SDARS Change History

Changes MOST FBlock SDARS 2.3 to MOST FBlock SDARS 2.4

| Change Ref. | Section | Changes |
|-------------|---------|---------------------------------------------|
| 2.4-1 | 3 | Added Function Block Dynamic Specification. |
| 2.4-2 | 2 | Translated some headings to English. |

1 Introduction

A MOST Function Catalog is a collection of MOST FunctionBlocks.

This document contains the specification of a FunctionBlock. MOST FunctionBlocks are standardized and maintained by MOST workgroup Device Architecture (WG_DA). In order to speed up the process of making new Function Blocks available, every Function Block will be updated individually as required.

The dynamic specification deals with application specific communication sequences. The current document concentrates on the exemplary sequences, it does not claim to be a complete description of the device communication over MOST.

2 FunctionBlock Definition

2.1 Definitions

Table 1 Text parameter data types

| | |
|--------------------|-----------|
| SHORTCATEGORYNAME: | String 8 |
| LongCategoryName: | String 36 |
| ShortStationName: | String 8 |
| LongStationName: | String 36 |
| Artist: | String 36 |
| AlbumName: | String 36 |
| AlbumLabel: | String 36 |
| InfoText0: | String 36 |
| InfoText1: | String 36 |
| InfoText2: | String 36 |
| InfoText3: | String 36 |
| InfoText4: | String 36 |
| Composer: | String 36 |
| ArtitsInfo1: | String 36 |
| ArtitsInfo2: | String 36 |
| Comments: | String 36 |
| | |

2.2 Introduction

Each function generally has the OPType Error and therefore the two parameters ErrorCode and ErrorInfo. The encoding of ErrorCode and ErrorInfo is described in MOST Specification in Section „Structure of MOST Protocols: OPType“, So the following lines must be regarded as part of the parameter listing of each function:

ErrorCode

(please refer to section „Structure of MOST Protocols: OPType“ in MOST Specification)

ErrorInfo

(please refer to section „Structure of MOST Protocols: OPType“ in MOST Specification)

2.3 SatRadio (FBlockID =0x44)

| Overview of functions | | |
|-----------------------|---------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| FktID-Codes | Function group | Functions |
| 0x000..0x002 | Coordination | FktIDs, Notification, NotificationCheck |
| 0x090..0x094 | Coordination for applications with ArrayWindows | CreateArrayWindow, DestroyArrayWindow, MoveArrayWindow, SearchArrayWindow, LongArrayInfo |
| 0x100..0x117 | Mandatory for applications with synchronous data source | SourceInfo, Allocate, DeAllocate, SourceConnect, SourceDisconnect, SourceRouting, SourceActivity, SourceName, SyncDataInfo |
| 0x210..0x217 | Mandatory functions of a SDARS Tuner | StationSelect, StationTune, CurrentStationInfo, CurrentCategoryInfo, RadioText, ServiceProvider, ElectronicCode, ServiceStatus |
| 0x410 ..0x415 | Optional functions of a SDARS Tuner | Presets, TuningStatus, StationListInfo, CategorySelect, CategoryListInfo, StationListforCategory |

2.4 Coordination

2.4.1 FktIDs (FktID = 0x000)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.2 Notification (FktID = 0x001)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.3 NotificationCheck (FktID = 0x002)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.4 CreateArrayWindow (FktID = 0x090)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.5 DestroyArrayWindow (FktID = 0x091)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.6 MoveArrayWindow (FktID = 0x092)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.7 SearchArrayWindow (FktID=0x093)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.8 LongArrayInfo (FktID=0x094)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.9 SourceInfo (FktID = 0x100)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.10 Allocate (FktID = 0x101)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.11 DeAllocate (FktID = 0x102)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.12 SourceConnect (0x105)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.13 SourceDisconnect (0x106)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.14 SourceRouting (0x107)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.15 SourceActivity (FktID = 0x103)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.16 SourceName (FktID = 0x104)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.4.17 SyncDataInfo (FktID = 0x116)

Please see Function Catalog GeneralFBlock (FblockID=0x00).

2.5 Mandatory Functions

2.5.1 StationSelect (FktID = 0x210)

By this function, the current Station can be displayed or a Station can be set.

Format of function

Function classes: unclassified property Number

| FBlock | Fkt | OPType | Parameter |
|----------|---------------|--------|----------------------|
| SatRadio | StationSelect | Set | Type, Station |
| | | SetGet | Type, Station |
| | | Get | Type |
| | | Status | Station |
| | | Error | ErrorCode, ErrorInfo |

Parameter

Type

The type of the current station

| Basis Datatype | Code | Description |
|----------------|-----------|----------------------------------------|
| Enum | 0x00 | Station Number |
| | 0x01 | SID |
| | 0x02-0xFF | Reserved for optional future offerings |

Station

The current station

Note: Station 0 is set aside by each Service Provider as an optional means of displaying the Electronic Code.

| Basis Datatype | Exp | Range of values | Step | Unit |
|----------------|-----|-----------------|------|-------------|
| Unsigned Byte | 0 | full range | 1 | not defined |

Method recommended

2.5.2 StationTune (FktID = 0x211)

This function allows tuning to stations within a set of station of a selected type.

Format of function

Function classes: Record of { Enumeration, Enumeration, Number}

| FBlock | Fkt | OPType | Parameter |
|----------|-------------|--------|----------------------|
| SatRadio | StationTune | Set | Pos, Data |
| | | SetGet | Pos, Data |
| | | Get | Pos |
| | | Status | Pos, Data |
| | | Error | ErrorCode, ErrorInfo |

Parameter

Pos

The parameter Pos={x,y} consists of two bytes x and y and shows which parameters shall be set, inquired or read. In case of Set only x = y = 0 is allowed.

| Datatype | Valid Range Pos X | Valid Range Pos Y |
|---------------|-------------------|-------------------|
| Unsigned Word | 0..3 | 0 |

Data

| PosX | PosY | Data |
|------|------|-----------------------------------|
| 0 | 0 | TuneDirection, ListType, Category |

TuneDirection

| Basis Datatype | Code | Description |
|----------------|------|-------------|
| Enum | 0x0 | Down |
| | 0x1 | Up |

ListType

The Type of StationSet within tuning is possible.

| Basis Datatype | Code | Description |
|----------------|------|--------------|
| Enum | 0x0 | Station |
| | 0x1 | Presets |
| | 0x2 | Category |
| | 0x3 | Favorite |
| | 0x8 | Unsubscribed |
| | 0xF | Unknown |

Category

The current category. If Category = 0 then navigation with up and down is done within the category list, thus selecting another category.

| Basis Datatype | Exp | Range of values | Step | Unit |
|----------------|-----|-----------------|------|-------------|
| Unsigned Byte | 0 | full range | 1 | not defined |

JKW: Category description is unclear????

SCAN is not realized????

If property TuneDirection needs an additional value e.g. neutral

Using a method is recommended

2.5.3 CurrentStationInfo (FktID = 0x212)

With this function the information about the currently tuned station may be asked.

Format of function

Function classes: Record of {Number, Text, Text, Enumeration, Number}

| Fblock | Fkt | OPType | Parameter |
|----------|--------------------|--------|----------------------|
| SatRadio | CurrentStationInfo | Get | Pos |
| | | Status | Pos, Data |
| | | Error | ErrorCode, ErrorInfo |

Parameter

Pos

The parameter Pos={x,y} consists of two bytes x and y and shows which parameters shall be set, inquired or read. Only x=y=0 is allowed.

| Basis Datatype | Valid Range Pos X | Valid Range Pos Y |
|----------------|-------------------|-------------------|
| Unsigned Word | 0 | 0 |

Data

| PosX | PosY | Data |
|------|------|------------------------------------------------------------------------------------------------|
| 0 | 0 | Station, ShortStationName, LongStationName, Subscription, Category, ActiveStatus, PresetNumber |

Station

The current station

Note: Station 0 is set aside by each Service Provider as an optional means of displaying the Electronic Code.

| Basis Datatype | Exp | Range of values | Step | Unit |
|----------------|-----|-----------------|------|-------------|
| Unsigned Byte | 0 | full range | 1 | not defined |

ShortStationName

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 8 |

LongStationName

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

Subscription

Defines the subscribed status for the given station.

| Basis Datatype | Code | Description |
|----------------|------|--------------|
| Enum | 0x0 | Unsubscribed |
| | 0x1 | Subscribed |

Category

The current category

| Basis Datatype | Exp | Range of values | Step | Unit |
|----------------|-----|-----------------|------|-------------|
| Unsigned Byte | 0 | full range | 1 | not defined |

CurrentCategoryInfo (FktID = 213)

This function gives information about the current Categories. Up to 24 different Categories are currently used.

Format of function

Function classes: Record of {Number, Text, Text}

| Fblock | Fkt | OPType | Parameter |
|----------|---------------------|--------|----------------------|
| SatRadio | CurrentCategoryInfo | Get | Pos |
| | | Status | Pos, Data |
| | | Error | ErrorCode, ErrorInfo |

Parameter

Pos

The parameter Pos={x,y} consists of two bytes x and y and shows which parameters shall be set, inquired or read. Only x=y=0 is allowed.

| Basis Datatype | Valid Range Pos X | Valid Range Pos Y |
|----------------|-------------------|-------------------|
| Unsigned Word | 0 | 0 |

Data

| PosX | PosY | Data |
|------|------|-----------------------------------------------|
| 0 | 0 | Category, ShortCategoryName, LongCategoryName |

Category

The current category information is asked.

| Basis Datatype | Exp | Range of values | Step | Unit |
|----------------|-----|-----------------|------|-------------|
| Unsigned Byte | 0 | full range | 1 | not defined |

ShortCategoryName

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 8 |

LongCategoryName

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

This property shows the various text fields associated with the current program selection. Controllers which cannot display all Radio Text data simply filter out only the usefull elements.

Function classes: Record of {Text, Text, Text, Text, Text, Text, Text, Text, Text, Text, Text, Text, Text, Text}

[illegible]

Pos

The parameter Pos={x,y} consists of two bytes x and y and shows which parameters shall be set, inquired or read.

| Basis Datatype | Valid Range Pos X | Valid Range Pos Y |
|----------------|-------------------|-------------------|
| Unsigned Word | 0..13 | 0 |

Data

| PosX | PosY | Data |
|------|------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | 0 | ProgramTitle, Artist, AlbumName, AlbumLabel, InfoText0, InfoText1, InfoText2, InfoText3, InfoText4, Composer, ArtitsInfo1, ArtitsInfo2, Comments |
| 1 | 0 | ProgramTitle |
| 2 | 0 | Artist |
| 3 | 0 | AlbumName |
| 4 | 0 | AlbumLabel |
| 5 | 0 | InfoText0 |
| 6 | 0 | InfoText1 |
| 7 | 0 | InfoText2 |
| 8 | 0 | InfoText3 |
| 9 | 0 | InfoText4 |
| 10 | 0 | Composer |
| 11 | 0 | ArtitsInfo1 |
| 12 | 0 | ArtitsInfo2 |
| 13 | 0 | Comments |

ProgramTitle

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

Artist

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

AlbumName

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

AlbumLabel

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

InfoText0

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

InfoText1

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

InfoText2

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

InfoText3

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

InfoText4

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

Composer

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

ArtistsInfo1

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

ArtistsInfo2

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

Comments

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

2.5.5 ServiceProvider (FktID = 0x215)

This function shows the information about the Service Provider and allows to change the Service Provider if available.

Format of function

Function classes: Enumeration

| FBlock | Fkt | OPType | Parameter |
|----------|-----------------|--------|----------------------|
| SatRadio | ServiceProvider | Set | ServiceProvider |
| | | SetGet | ServiceProvider |
| | | Get | |
| | | Status | ServiceProvider |
| | | Error | ErrorCode, ErrorInfo |

Parameter

ServiceProvider

| Basis Datatype | Code | Description |
|----------------|------|-------------|
| Enum | 0x0 | XM |
| | 0x1 | Sirius |
| | 0xFF | Other |

2.5.6 ElectronicCode (FktID = 0x216)

With this function the Electronic code of the module can be retrieved.

Format of function

Function classes: Text

| FBlock | Fkt | OPType | Parameter |
|----------|----------------|--------|----------------------|
| SatRadio | ElectronicCode | Get | |
| | | Status | ElectronicCode |
| | | Error | ErrorCode, ErrorInfo |

Parameter

ElectronicCode

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 12 |

2.5.7 ServiceStatus (FktID = 0x217)

This function gives information about the condition of the entire service, but does not give any information about individual stations and their subscription status.

Format of function

Function classes: Record of {Switch, Switch, Enumeration}

| FBlock | Fkt | OPType | Parameter |
|----------|---------------|--------|----------------------|
| SatRadio | ServiceStatus | Get | Pos |
| | | Status | Pos, Data |
| | | Error | ErrorCode, ErrorInfo |

Parameter

Pos

The parameter Pos={x,y} consists of two byte x and y and shows which parameter shall be set, inquired or read.

| Basis-Datatype | Valid Range Pos X | Valid Range Pos Y |
|----------------|-------------------|-------------------|
| Unsigned Word | 0..4 | 0 |

Data

| PosX | PosY | Data |
|------|------|------------------------------------------------------------------|
| 0 | 0 | SignalStatus, AudioStatus, StationListStatus, SubscriptionStatus |
| 1 | 0 | AudioStatus |
| 2 | 0 | StationListStatus |
| 3 | 0 | SubscriptionStatus |

AudioStatus

This parameter defines the availability of the service through audio availability.

| Basis Datatype | True-Description | False-Description |
|----------------|------------------|---------------------|
| Bool | Audio available | Audio not available |

StationListStatus

This parameter defines the status of broadcast station list.

| Basis Datatype | True-Description | False-Description |
|----------------|---------------------------------------------|----------------------------------------------|
| Bool | Station List is current, no update required | Station List is out of date; update required |

SubscriptionStatus

This parameter defines if the Subscription Status of the entire Service is activated or not.

| Basis Datatype | Code | Description |
|----------------|------|---------------|
| Enum | 0x0 | Not activated |
| | 0x1 | Activated |

2.6 Optional Functions

2.6.1 Presets (FktID = 0x410)

Format of function

Function classes: Array [1..24] of Record of {Enumeration, Number, Text, Text, Enumeration}

| FBlock | Fkt | OPType | Parameter |
|----------|---------|--------|----------------------|
| SatRadio | Presets | Set | Pos, Data |
| | | SetGet | Pos, Data |
| | | Get | Pos |
| | | Status | Pos, Data |
| | | Error | ErrorCode, ErrorInfo |

Parameter

Pos

The parameter Pos={x,y} consists of two byte x and y and shows which parameter shall be set, inquired or read.

| Basis Datatype | Valid Range Pos X | Valid Range Pos Y |
|----------------|-------------------|-------------------|
| Unsigned Word | 0..24 | 0..5 |

Data

| PosX | PosY | Data |
|------|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | 0 | PresetSelection[1], Station[1], ShortStationName[1], LongStationName[1], Subscription[1], PresetSelection[2], Station[2], ShortStationName[2], LongStationName[2], Subscription[2],...,PresetSelection[24], Station[24], ShortStationName[24], LongStationName[24], Subscription[24] |
| x | 0 | PresetSelection[x], Station[x], ShortStationName[x], LongStationName[x], Subscription[x] |
| x | 1 | PresetSelection[x] |
| x | 2 | Station[x] |
| x | 3 | ShortStationName[x] |
| x | 4 | LongStationName[x] |
| x | 5 | Subscription[x] |

PresetSelection

| Adr | Description | Type | Comment |
|-----|-------------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0x0 | Inactive | Status | No value of station number has been associated with the indicated preset. |
| 0x1 | Active | Status | Station number value has been associated with an indicated preset |
| 0x2 | Select | Process | Tunes the SDARS tuner to a station that corresponds to a preset For Example station 38 is "stored" preset 3. If the Preset 3 "button" is selected then the tuner will tune to station 38. The message sent to the tuner would be: Preset.set(3,1,2) |
| 0x3 | Store | Process | Storing the current station in the indicated preset "button". For example if you wanted to store the current station in preset 4 the message sent to the current station in preset 4 the message sent to the tuner would be: Preset.set(4,1,3). |

| Basis Datatype | Code | Description |
|----------------|------|-------------|
| Enum | 0x0 | no active |
| | 0x1 | Active |
| | 0x2 | Select |
| | 0x3 | Set |

Station

The current station

Note: Station 0 is set aside by each Service Provider as an optional means of displaying the Electronic Code.

| Basis Datatype | Exp | Range of values | Step | Unit |
|----------------|-----|-----------------|------|-------------|
| Unsigned Byte | 0 | full range | 1 | not defined |

ShortStationName

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 8 |

LongStationName

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

Subscription

Defines the subscribed status for the given station.

| Basis Datatype | Code | Description |
|----------------|------|--------------|
| Enum | 0x0 | Unsubscribed |
| | 0x1 | Subscribed |

2.6.2 TuningStatus (FktID = 0x411)

This function displays the current tuning status.

Format of function

Function classes: Enumeration

| Fblock | Fkt | OPType | Parameter |
|----------|---------------|--------|----------------------|
| SatRadio | StationSelect | Get | |
| | | Status | TuningStatus |
| | | Error | ErrorCode, ErrorInfo |

Parameter

TuningStatus

The current tuning status to inform the controller if a station selection is complete

| Basis Datatype | Code | Description |
|----------------|------|---------------------|
| Enum | 0x0 | Tuning |
| | 0x1 | Tuned |
| | 0x2 | Unsuccessful tuning |

2.6.3 StationListInfo (FktID = 0x412)

This property shows the various text fields associated with station's short station name.

This data structure can be accessed directly or by means of an array window. The dynamic array windows can be in the range 0x413 to 0x41F

Format of function

Function classes: DynamicArray [1..N (>=0)] of Record of {Tag: Number
Station: Number
ShortStationName: Text
LongStationName: Text
Subscription: Enumeration
Category: Number}

| FBlock | Fkt | OPType | Parameter |
|----------|-------------------------|--------|----------------------|
| SatRadio | ShortCurrentStationInfo | Get | Tag, PosY |
| | | Status | Tag, PosY, Data |
| | | Error | ErrorCode, ErrorInfo |

Parameter

Tag

This parameter is a key indicating a selected record in a dynamic array. Its range is from 0x0 to 0xFFFF. Tag = 0x0 means select the whole array; Tag = 0xFFFF indicates the end of the dynamic array

| Basis Datatype | Valid Range Tag |
|----------------|-----------------|
| Unsigned Word | 0..N (>=0) |

PosY

The parameter PosY consists of one byte and shows what is to be set, requested or read in the record.

| Basis Datatype | Valid Range PosY |
|----------------|------------------|
| Unsigned Byte | 0..4 |

Data

| Tag | PosY | Data |
|-----|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | 0 | Tag[1], Station[Tag[1]], ShortStationName[Tag[1]], Subscription[Tag[1]], LongStationName[Tag[1]], Category[Tag[1]], Tag[N], Station[Tag[N]], ShortStationName[Tag[N]], Subscription[Tag[N]], LongStationName[Tag[N]], Category[Tag[N]] |
| Tag | 0 | Tag[N], Station[Tag[x]], ShortStationName[Tag[x]], Subscription[Tag[x]], LongStationName[Tag[X]], Category[Tag[X]] |
| Tag | 1 | Station[Tag[x]] |
| Tag | 2 | ShortStationName[Tag[x]] |
| Tag | 3 | Subscription[Tag[x]] |
| Tag | 4 | LongStationName[Tag[X]] |
| Tag | 4 | Category[Tag[X]] |

Station

The current station

Note: Station 0 is set aside by each Service Provider as an optional means of displaying the Electronic Code.

| Basis Datatype | Exp | Range of values | Step | Unit |
|----------------|-----|-----------------|------|-------------|
| Unsigned Byte | 0 | full range | 1 | not defined |

ShortStationName

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 8 |

Subscription

Defines the subscribed status for the given station.

| Basis Datatype | Code | Description |
|----------------|------|--------------|
| Enum | 0x0 | Unsubscribed |
| | 0x1 | Subscribed |

LongStationName

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 20 |

Category

The current category

| Basis Datatype | Exp | Range of values | Step | Unit |
|----------------|-----|-----------------|------|-------------|
| Unsigned Byte | 0 | full range | 1 | not defined |

2.6.4 CategorySelect (FktID = 0x413)

This function displays and sets the Category.

Format of function

Function classes: Number

| Fblock | Fkt | OPType | Parameter |
|----------|----------------|--------|----------------------|
| SatRadio | CategorySelect | Set | Category |
| | | SetGet | Category |
| | | Get | |
| | | Status | Category |
| | | Error | ErrorCode, ErrorInfo |

Parameter

Category

The current category

| Basis Datatype | Exp | Range of values | Step | Unit |
|----------------|-----|-----------------|------|-------------|
| Unsigned Byte | 0 | full range | 1 | not defined |

2.6.5 CategoryListInfo (FktID = 0x414)

Format of function

Function classes: **array** of Record of {Category : Number, ShortCategoryName : Text, LongCategoryName : Text}

| FBlock | Fkt | OPType | Parameter |
|----------|-------------------|--------|----------------------|
| SatRadio | ShortCategoryInfo | Get | Pos |
| | | Status | Pos, Data |
| | | Error | ErrorCode, ErrorInfo |

Parameter

Pos

The parameter Pos={x,y} consists of two byte x and y and shows which parameter shall be set, inquired or read.

| Basis Datatype | Valid Range Pos X | Valid Range Pos Y |
|----------------|-------------------|-------------------|
| Unsigned Word | 0 | 0 |

Data

| PosX | PosY | Data |
|------|------|------------------------------------------------------------------------------------------------------------|
| 0 | 0 | Category[1], ShortCategoryName[1], Category[2], ShortCategoryName[2],...,Category[N], ShortCategoryName[N] |

Category

The current category

| Basis Datatype | Exp | Range of values | Step | Unit |
|----------------|-----|-----------------|------|-------------|
| Unsigned Byte | 0 | full range | 1 | not defined |

ShortCategoryName

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 8 |

LongCategoryName

| Basis Datatype | MaxSize |
|----------------|---------|
| String | 36 |

2.6.6 StationListforCategory (FktID = 0x415)

Format of function

Function classes: Unclassified Method

| Fblock | Fkt | OPType | Parameter |
|----------|---------------------|----------------|------------------------------------|
| SatRadio | GetCategoryStations | StartResultAck | SenderHandle, Category |
| | | ResultAck | SenderHandle, StationList |
| | | ErrorAck | SenderHandle, ErrorCode, ErrorInfo |

Parameter

SenderHandle

ID of the sender process within one device

| Basis Datatype | Range of values |
|----------------|-----------------|
| Unsigned Word | full range |

Category

The current category

| Basis Datatype | Exp | Range of values | Step | Unit |
|----------------|-----|-----------------|------|-------------|
| Unsigned Byte | 0 | full range | 1 | not defined |

StationList

List of Stations for the current category

| Basis Datatype | Max.Length in Byte | Description |
|----------------|--------------------|---------------------------------------------|
| Stream | 1024 | StationList ::= (Station, ShortStationName) |

Station

Station in the Station List Stream

Note: Station 0 is set aside by each Service Provider as an optional means of displaying the Electronic Code.

| Basis Datatype | Exp | Range of values | Step | Unit |
|----------------|-----|-----------------|------|-------------|
| Unsigned Byte | 0 | 0..255 | 1 | not defined |

ShortStationName

Station Name in the Station List Stream

| Basis Datatype | MaxSize |
|----------------|---------|
| String | - |

3 FunctionBlock Dynamic Specification

3.1 Startup

3.1.1 Notification

The notification for a set of functions for an application is implementation specific.

3.2 Activating the Satellite Radio application

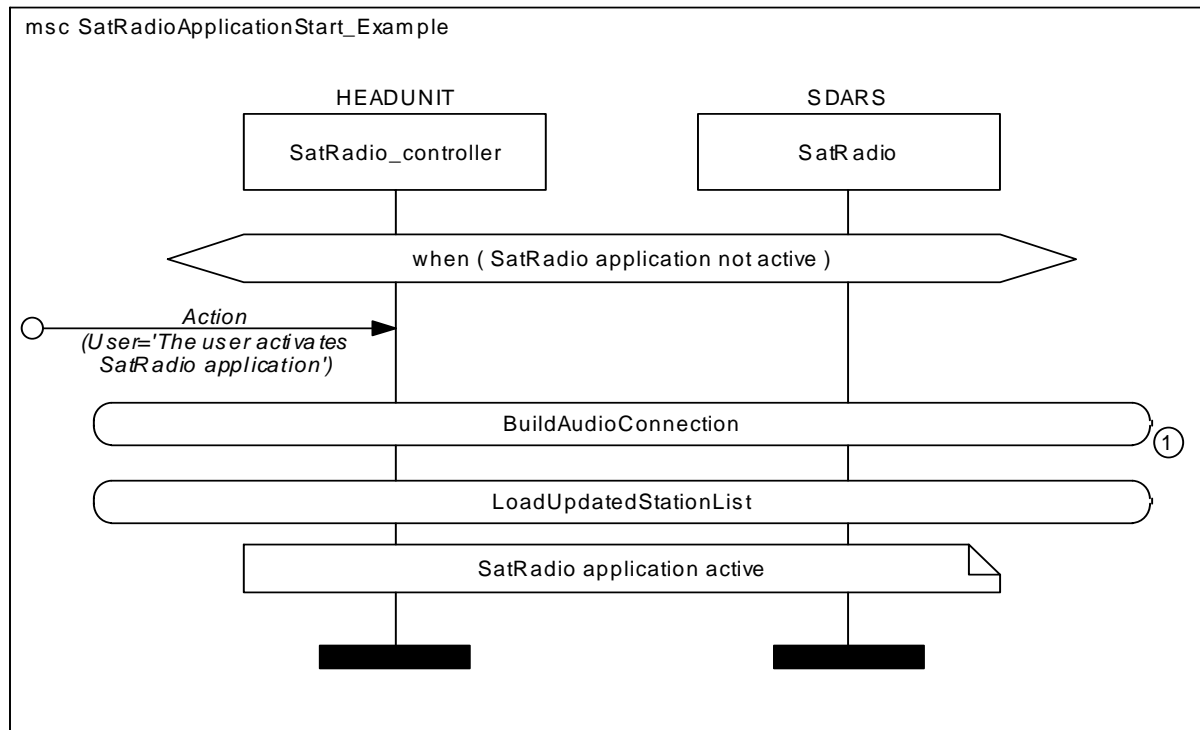
| | | | |
|------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|---------|
| Use Case: | Satellite radio application activation | | |
| Description: | Activation of the satellite radio application initiates an audio connection between the source and the sink. Before any user action the satellite radio is expected to send an update of the (all) station list. | | |
| Prior Condition: | | | |
| Initiator: | Passenger | Internal | Comment |
| | X | | |
| MMI Events | | | |
| Remarks: | This procedure is implementation specific | | |

3.2.1.1 General MSC's

Not available!

3.2.1.2 Example MSC's

SDARS_MOSTCo



Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 1: Satellite radio application activation

- 1 The audio handling is not part of the SDARS communication, thus not further described here.

3.2.1.3 Error MSC's

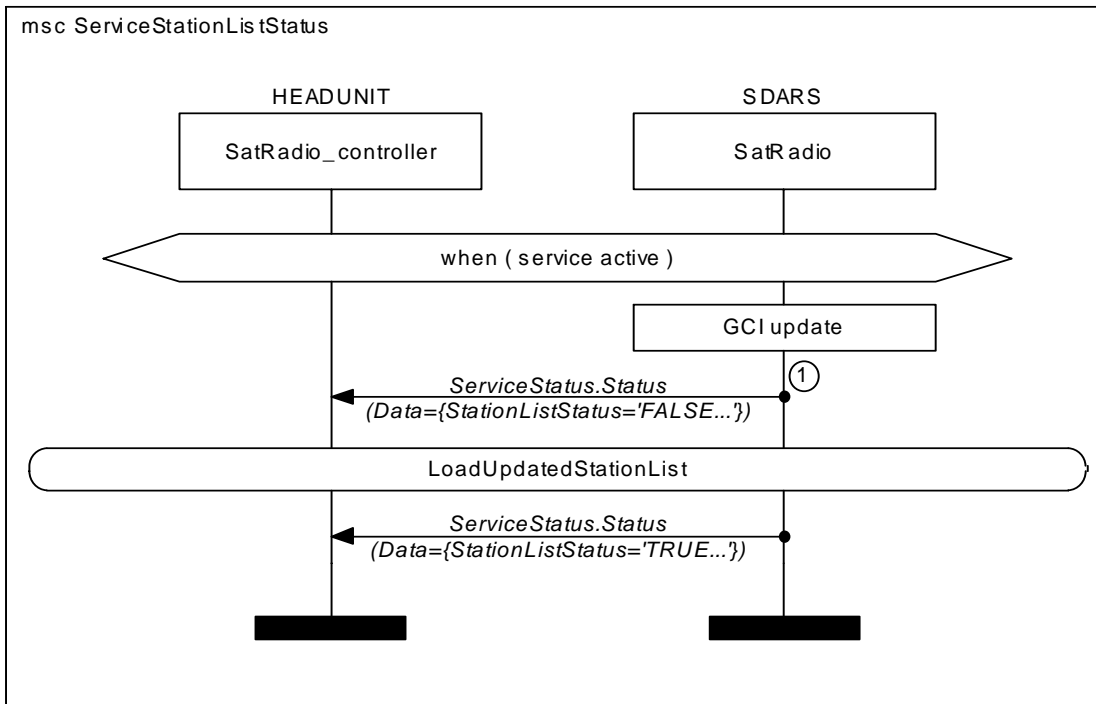
Not available!

3.3 Service Signal Acquisition

| | | | |
|------------------|------------------------------------------------------------------------------------|----------|---------|
| Use Case: | Service Activation and Deactivation | | |
| Description: | The SDARS module will be activated and deactivated via its service status function | | |
| Prior Condition: | | | |
| Initiator: | Passenger | Internal | Comment |
| | X | | |
| MMI Events | | | |
| Remarks: | | | |

3.3.1.1 General MSC's

SDARS_MOSTCo



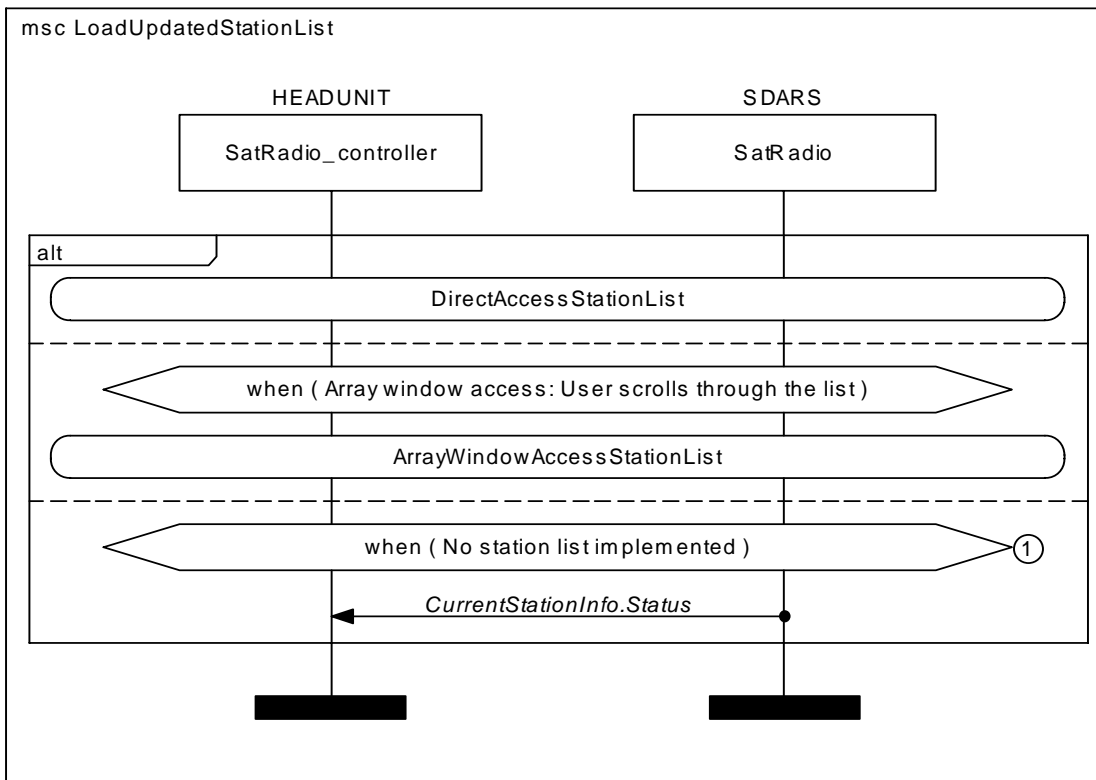
Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 2: Service station list update indication

1 Station list is out of date, update required

SDARS_MOSTCo



Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 3: Service station list update indication

1 the list contains only one station

3.3.1.2 Example MSC's

Not Available

3.3.1.3 Error MSC's

Not Available

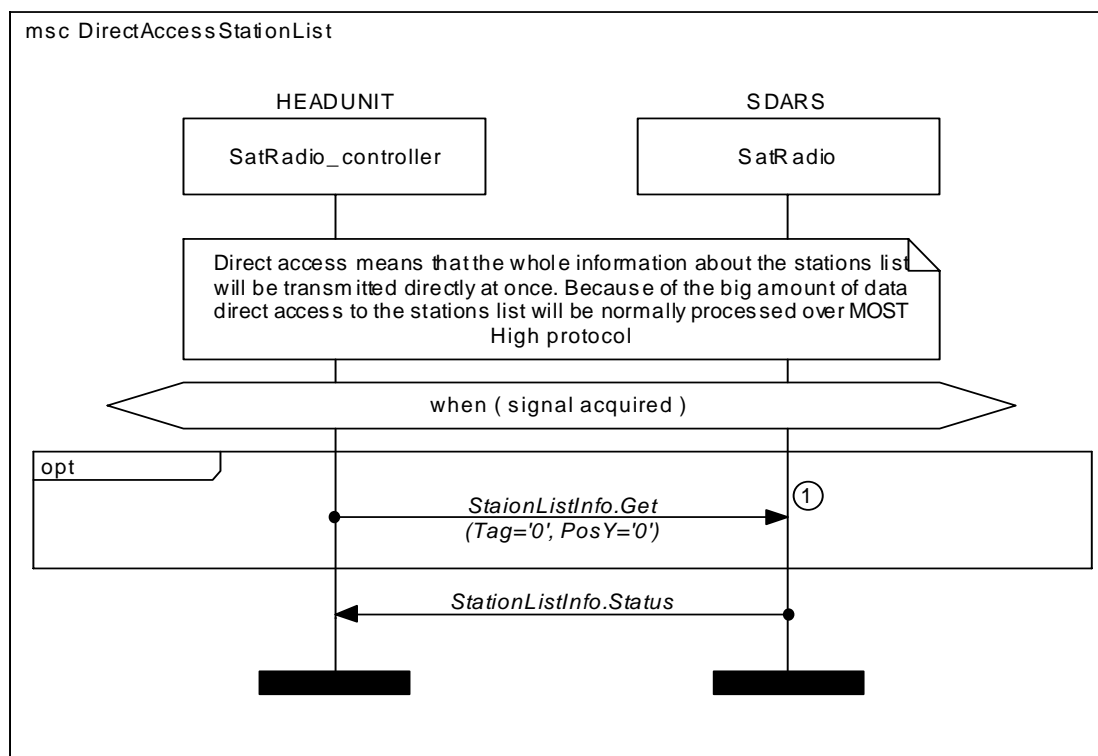
3.4 Station Selection

3.4.1 Direct Access to the Station List

| | | | |
|------------------|------------------------------------------------------------------------------------------------------------------------------------------------|----------|---------|
| Use Case: | Selecting the station list | | |
| Description: | Direct access means here that when the user selects the SDARS station list menu, the SatRadio transmits the whole list of station information. | | |
| Prior Condition: | | | |
| Initiator: | Passenger | Internal | Comment |
| | X | | |
| MMI Events | | | |
| Remarks: | | | |

3.4.1.1 General MSC's

SDARS_MOSTCo



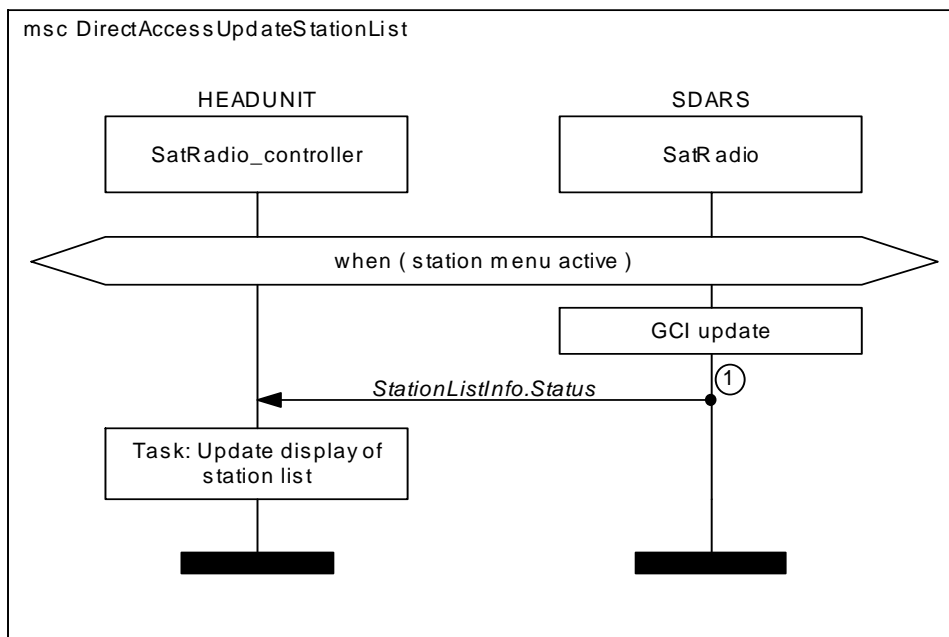
Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 4: Selecting the whole station list

- 1 The controller requests the whole List

SDARS_MOSTCo



Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 5: Update station list when Notification for function activated

- 1 Caution: Everytime - if the controller is notified on this function - some of the parameters change the complete data will be transmitted.

3.4.1.2 Example MSC's

Not available

3.4.1.3 Error MSC's

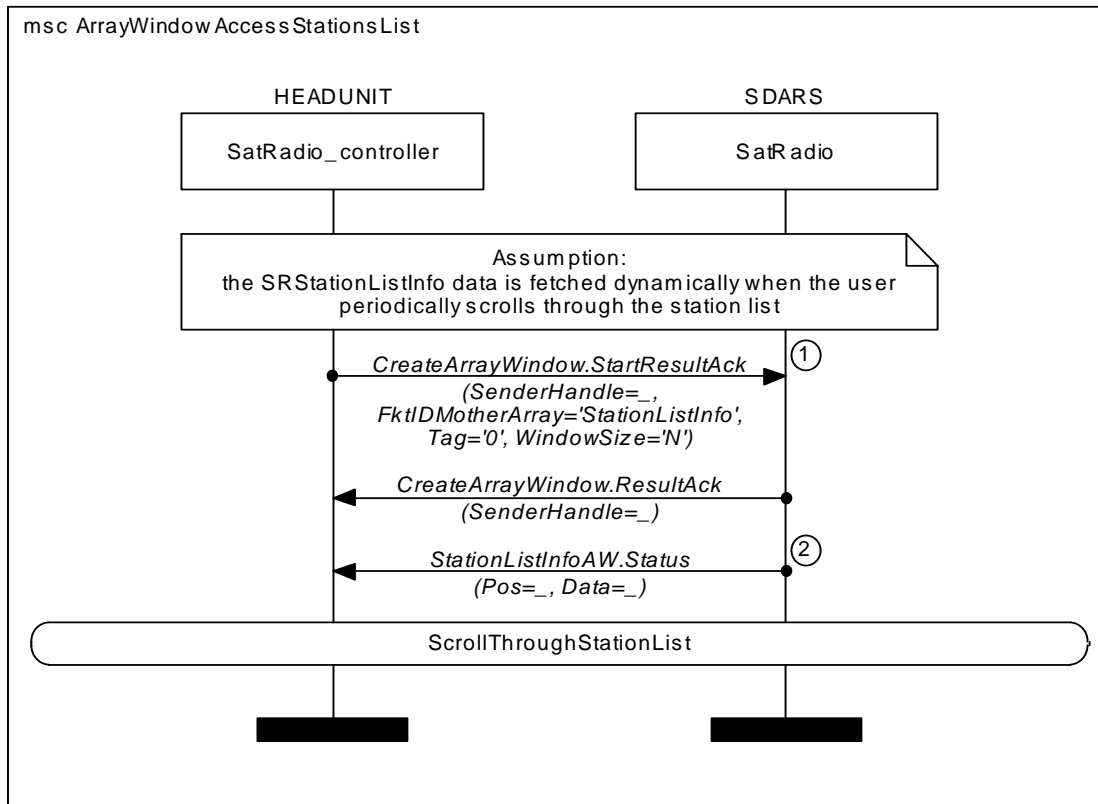
Not available

3.4.2 Selecting the Station List using Array Windows

| | | | |
|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------|----------|---------|
| Use Case: | Selecting the station list | | |
| Description: | Alternative to the direct access, the array window mechanism is used to transfer the stations list piecewise on demand, avoiding high bus load. | | |
| Prior Condition: | | | |
| Initiator: | Passenger | Internal | Comment |
| | | X | |
| MMI Events | | | |
| Remarks: | | | |

3.4.2.1 General MSC's

SDARS_MOSTCo



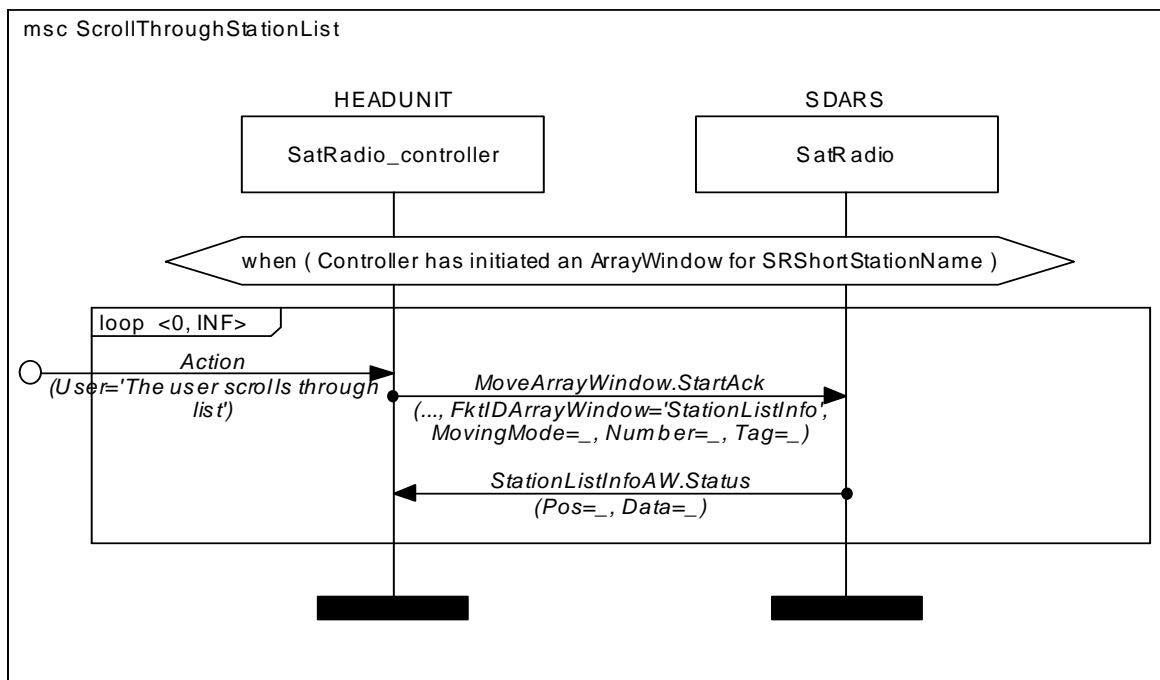
Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 6: Transmitting the SDARS station menu to the controller using Mother Arrays

- 1 The array window can be initialised during startup independent of the user request
The parameter FktIDMotherArray takes the FktID of StationListInfo; Tag is the index of the element to start with, 0 indicating to start from the beginning; WindowSize is the size of the array window, depending on the number of display lines
- 2 First window fetched

SDARS_MOSTCo



Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 7: Scrolling through an array window

3.4.2.2 Exampel MSC's

Not Available

3.4.2.3 Error MSC's

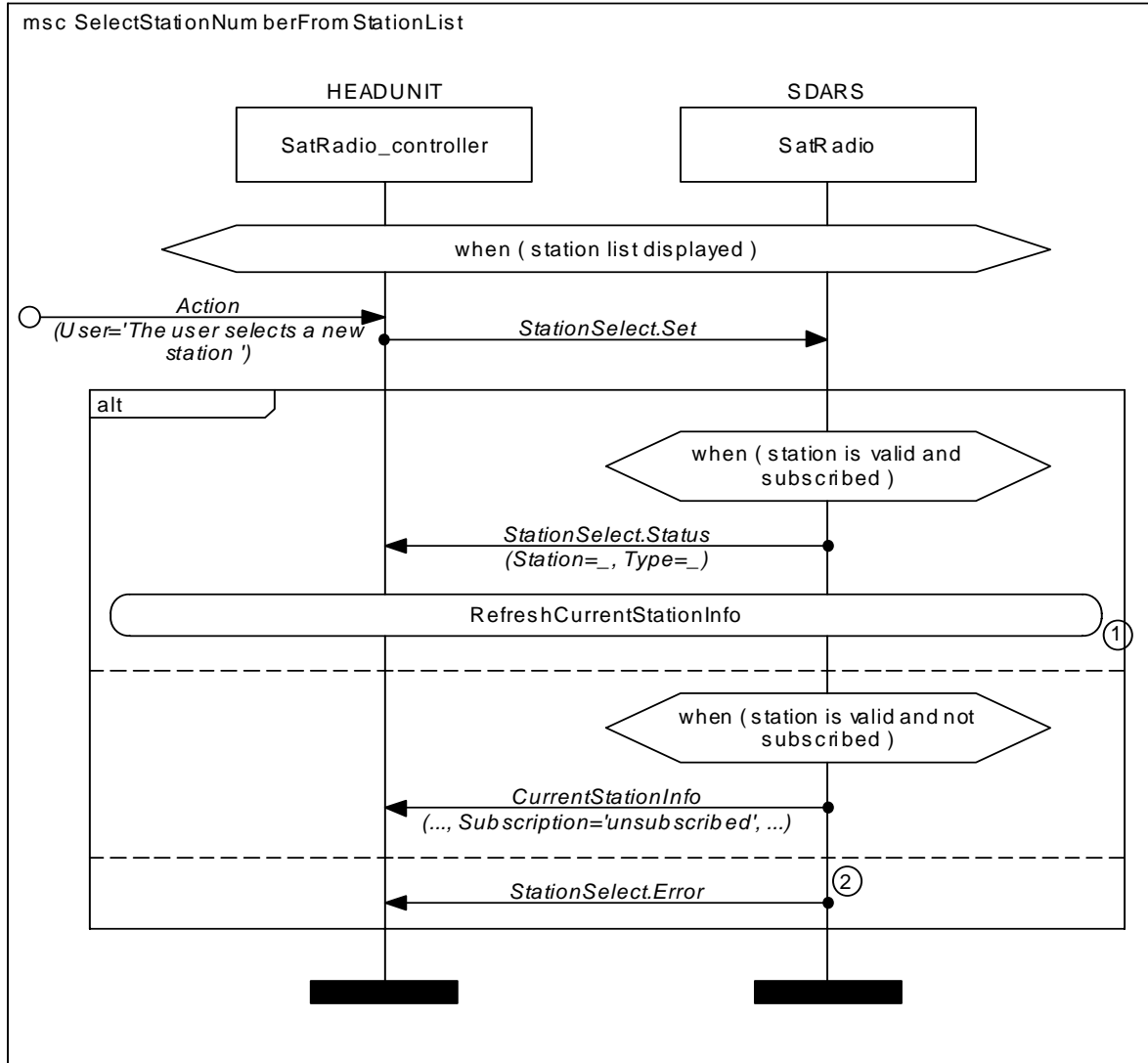
Not Available

3.4.3 Select Station from Station List

| | | | |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------|----------|---------|
| Use Case: | Select station number (subscribed) | | |
| Description: | When the user selects a particular station on the MMI display, SDARS starts a tuning procedure and informs the MMI about its new state. | | |
| Prior Condition: | | | |
| Initiator: | Passenger | Internal | Comment |
| | X | | |
| MMI Events | | | |
| Remarks: | | | |

3.4.3.1 General MSC's

SDARS_MOSTCo



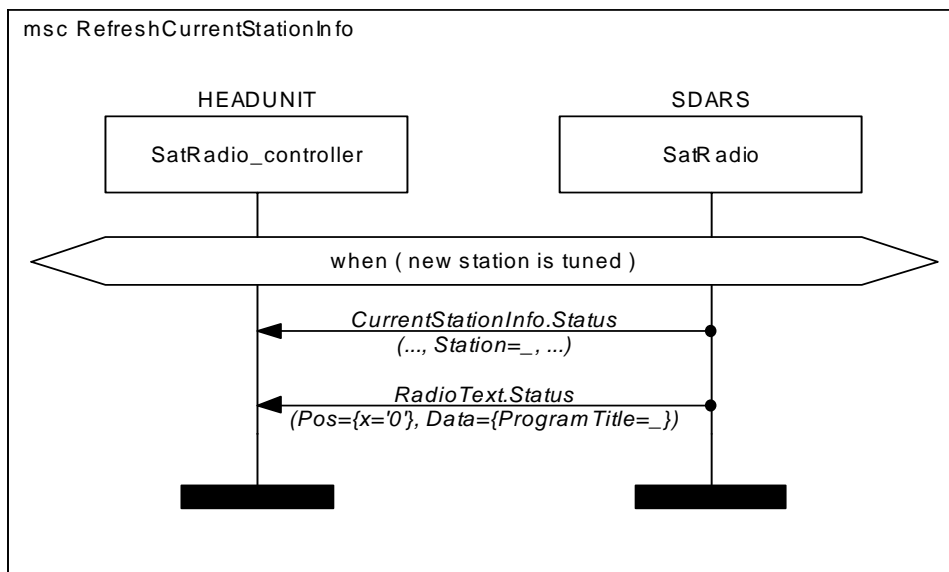
Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 8: Select station from the station menu

- 1 SDARS is tuned to the selected station. Station information is transferred to the controller
- 2 Invalid stations only

SDARS_MOSTCo



Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 9: Refresh current station info

3.4.3.2 Example MSC's

Not Available

3.4.3.3 Error MSC's

See section General MSC's

3.4.4 Select Station by Category

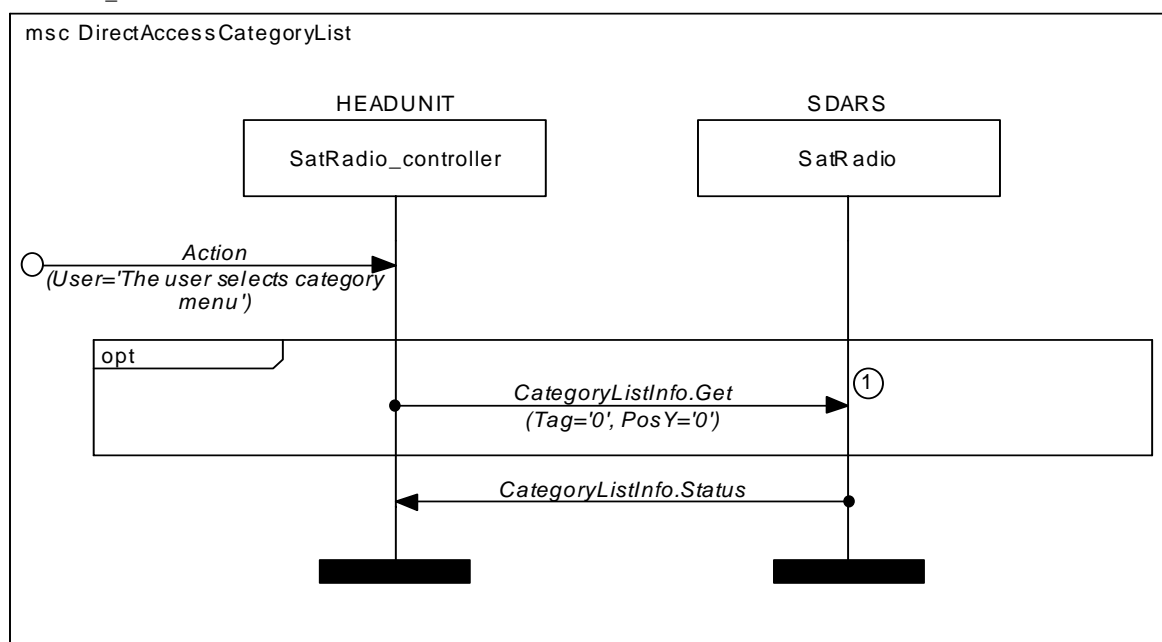
| | | | |
|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|---------|
| Use Case: | Select station by Category | | |
| Description: | When the user selects the category menu the category information for all categories will be transmitted to the MMI. Choosing a category will cause all stations for this category to be fetched and displayed. | | |
| Prior Condition: | | | |
| Initiator: | Passenger | Internal | Comment |
| | X | | |
| MMI Events | | | |
| Remarks: | | | |

3.4.4.1 Direct Access to the Category List

| | | | |
|------------------|---------------------------------------------------------|----------|---------|
| Use Case: | Direct access to the category list | | |
| Description: | The whole category list is transmitted with one message | | |
| Prior Condition: | | | |
| Initiator: | Passenger | Internal | Comment |
| | X | | |
| MMI Events | | | |
| Remarks: | | | |

3.4.4.1.1 General MSC's

SDARS_MOSTCo



Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 10: Direct access to the category list

1 The controller requests the whole List

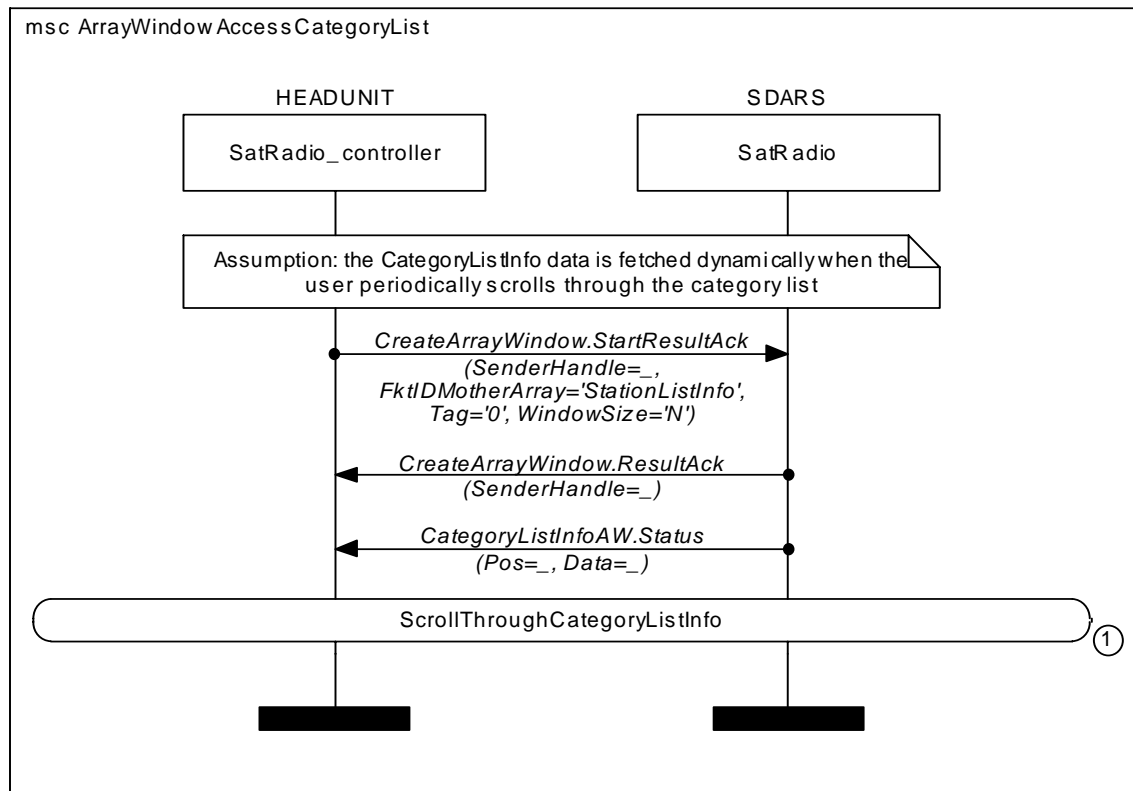
3.4.4.1.2 Example MSC's

3.4.4.1.3 Error MSC's

3.4.4.2 Array Window Access to the Category List

3.4.4.2.1 General MSC's

SDARS_MOSTCo



Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 11: Load category list using Array Windows

1 same as ScrollThroughStationListInfo

3.4.4.2.2 Example MSC's

3.4.4.2.3 Error MSC's

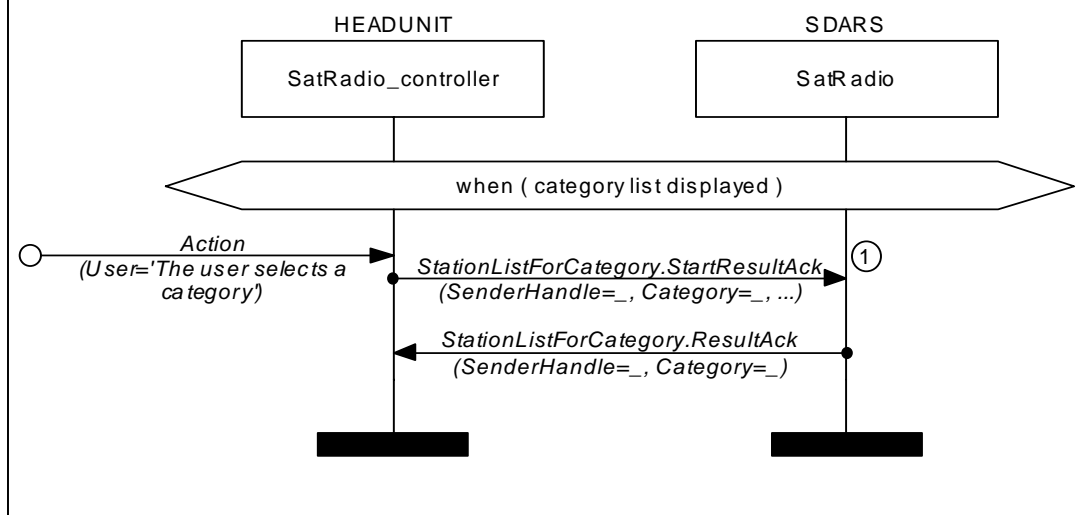
3.4.4.3 Loading the Station List for a selected Category

| | | | |
|------------------|--------------------------------------------------------------------------------------------------------|----------|---------|
| Use Case: | Select a station by category | | |
| Description: | Choosing from the category list the user selects a category for which the stations are to be displayed | | |
| Prior Condition: | | | |
| Initiator: | Passenger X | Internal | Comment |
| MMI Events | | | |
| Remarks: | | | |

3.4.4.3.1 General MSC's

SDARS_MOSTCo

msc SelectStationByCategory



Revision: 1.2 Date: 2004/05/27 12:00:10

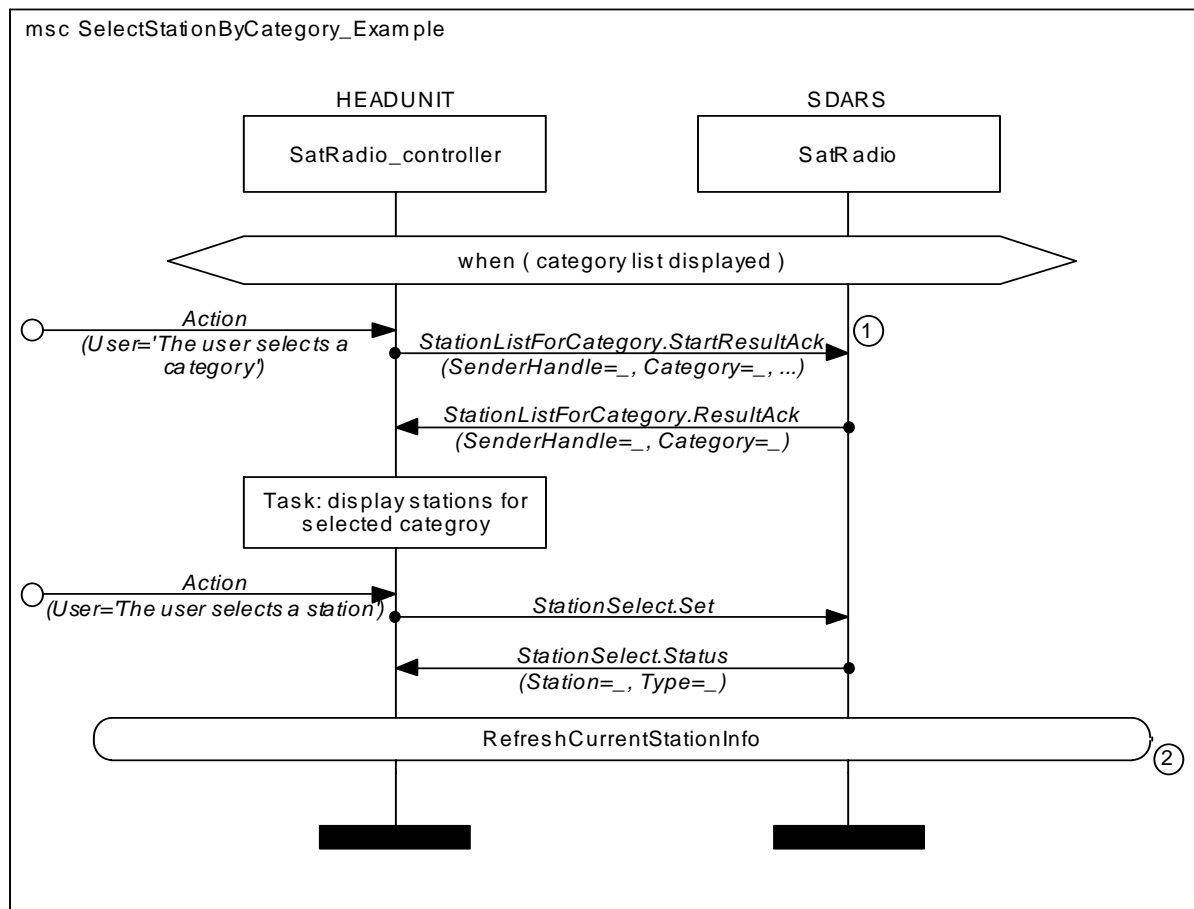
SDARS_MOSTCo_V1_0.mpr,v

MSC 12: Get station list by category

- 1 select all available stations for a particular category i

3.4.4.3.2 Example MSC's

SDARS_MOSTCo



Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 13: Select a station by category example

- 1 select all available stations for a particular category i
- 2 SDARS is tuned to the selected station. Channel information is transferred to the controller

3.4.5 Scanning Through Stations

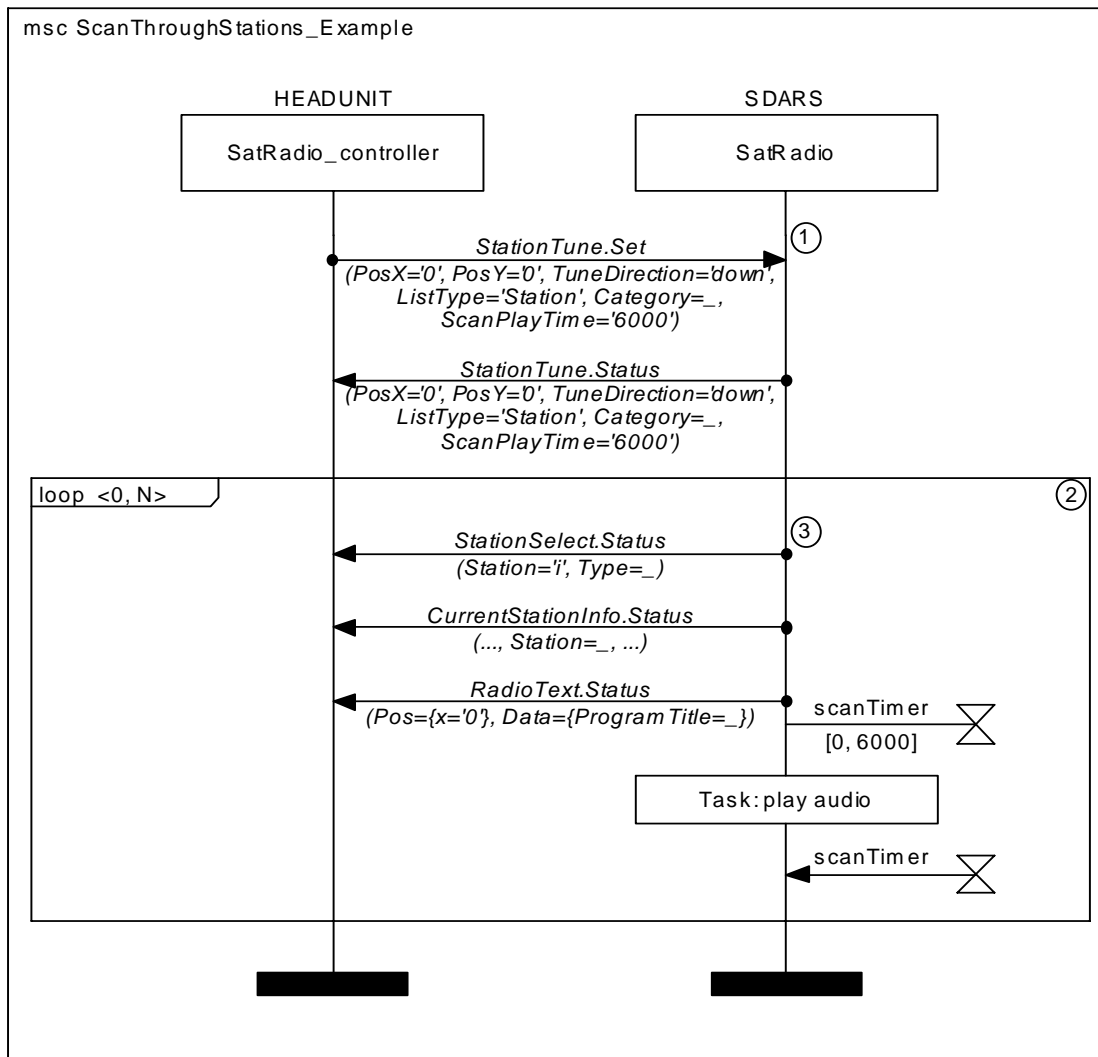
| | | | |
|------------------|-------------------------------------------------------------------------|----------|---------|
| Use Case: | Scan through a station | | |
| Description: | The user can activate a scan function similar to a scan in a CD player. | | |
| Prior Condition: | | | |
| Initiator: | Passenger | Internal | Comment |
| | X | | |
| MMI Events | | | |
| Remarks: | | | |

3.4.5.1 General MSC's

Not available!

3.4.5.2 Example MSC's

SDARS_MOSTCo



Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 14: Select a station by category example

- 1 If scanPlayTime > 0 then the tuner performs a scan procedure
- 2 loop through the currently scanned list. There is a maximum of N stations in a list
- 3 the tuner loops to the next station to play for a ScanPlayTime period

3.4.5.3 Error MSC's

Not available!

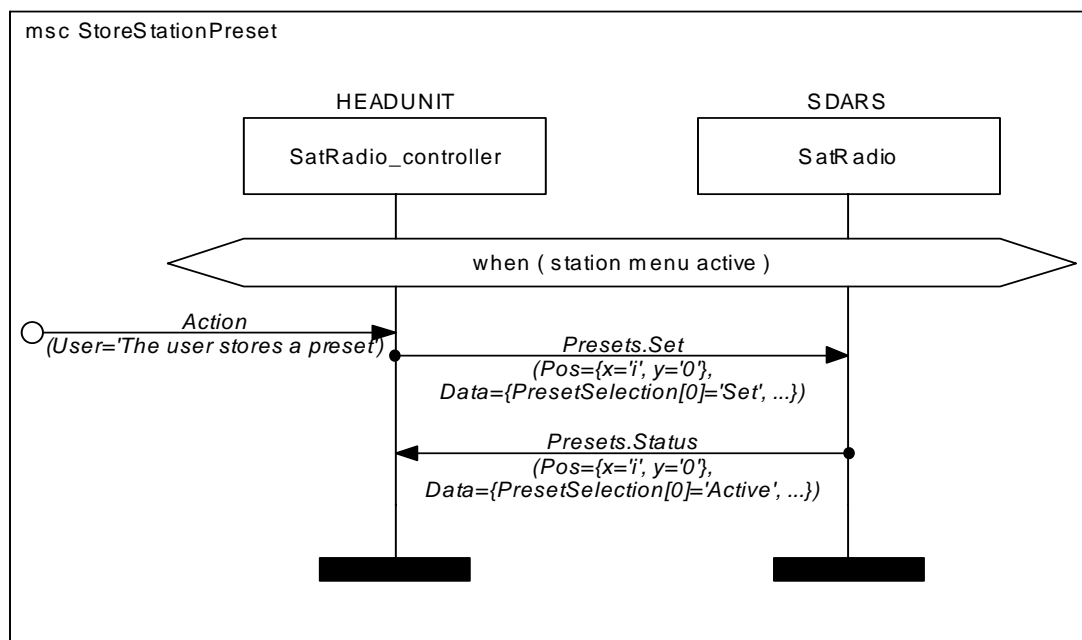
3.4.6 Station Preset Menu

3.4.6.1 Store Preset Menu

| | | | |
|------------------|------------------------------------|----------|---------|
| Use Case: | Store station preset by category | | |
| Description: | The user can store station presets | | |
| Prior Condition: | | | |
| Initiator: | Passenger | Internal | Comment |
| | X | | |
| MMI Events | | | |
| Remarks: | | | |

3.4.6.1.1 General MSC's

SDARS_MOSTCo



Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 15: Store station preset

3.4.6.1.2 Example MSC's

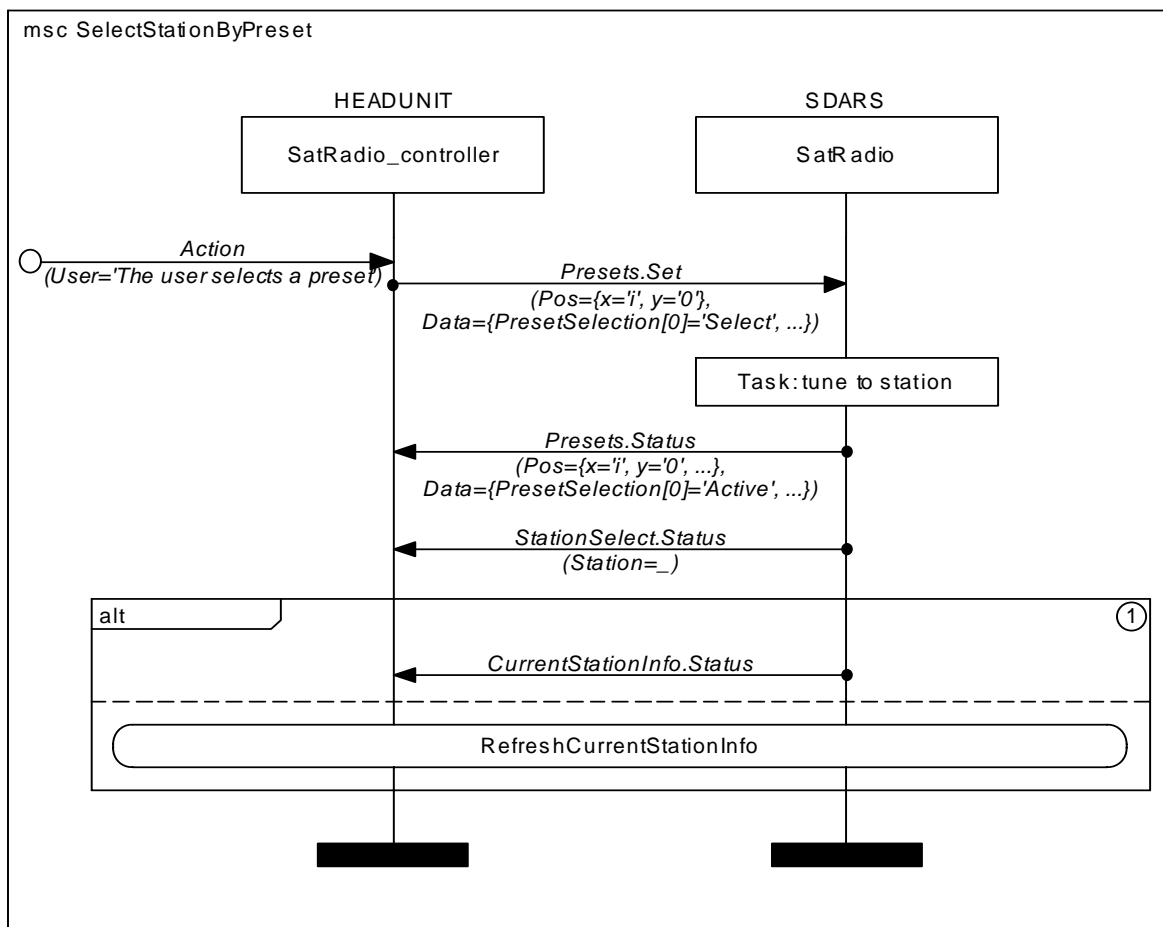
3.4.6.1.3 Error MSC's

3.4.6.2 Select Station Preset

| | | | |
|------------------|-------------------------------------|----------|---------|
| Use Case: | Select station preset by category | | |
| Description: | The user can select station presets | | |
| Prior Condition: | | | |
| Initiator: | Passenger | Internal | Comment |
| | X | | |
| MMI Events | | | |
| Remarks: | | | |

3.4.6.2.1 General MSC's

SDARS_MOSTCo



Revision: 1.2 Date: 2004/05/27 12:00:10

SDARS_MOSTCo_V1_0.mpr,v

MSC 16: Select station by preset

1 Depending on the implementation

3.4.6.2.2 Example MSC's

Not Available

