

Supplementary release documentation for version 1.72

1.1 Querying information on the keep file

Running a cleanup card task will clean the card from all content that is not used by any of the playlists available on the card. When this is not desired a keep file (named keep.me) is added to the content update file, containing all of the files which are not to be deleted even when they are not used by any of the playlists.

Sometimes, it is required that you make sure that this file is indeed on the card before the cleanup is issued. For this purpose a new command was added in version 1.72, which can be used to request information about the keep file.

Based on the information returned, it is possible to validate the presence of the keep file and whether this is the recent one or not.

Description: Check whether keep.me is present on the card and when it was created/updated

Devices: SIR100/SIR120/SIR120PRO/SIR150

Command: chkme

Example 1: Keep file available

Srv→Dev: chkme\n

Dev→Srv: \r\nFilesize:\t2956 bytes\r\nCreated:\t12:22 on 10/04/2015\r\n>

Example 2: Keep file not available (or no card present)

Srv→Dev: chkme\n

Dev→Srv: \r\nNOFILE\r\n>

1.2 Deleting a single file from card

The most used option to clean card content is using the card cleanup task. The cleanup will start by first compiling the information on what has to be kept, information which is again written first to the card. Should the card have run full before we had the chance to clean up, then the cleanup will fail.

For the situation mentioned above, but also for those cases when you want to remove a few specific (known) files without the need to setup a keep file, a delete command was added. This command can be used to delete a file from card the filename of which is known.

Description: Check whether keep.me is present on the card and when it was created/updated

Devices: SIR100/SIR120/SIR120PRO/SIR150

Command: del <filename>

Example 1: Delete file t123456.mp3 - file present on the card

Srv→Dev: del t123456.mp3\n

```
Dev→Srv: \r\nOK\r\n>
```

Example 2: Delete file t123456.mp3 - file not present on the card (or no card present)

```
Srv→Dev: del t123456.mp3\r\n
```

```
Dev→Srv: \r\nNOFILE\r\n>
```

Example 3: Delete file t123456.mp3 - Issue with (writing to) card

```
Srv→Dev: del t123456.mp3\r\n
```

```
Dev→Srv: \r\nFAILED\r\n>
```

Example 3: Delete command without argument

```
Srv→Dev: del\r\n
```

```
Dev→Srv: \r\nERROR\r\n>
```

! The first time (after a reset) that the SIR needs to write to the card (deleting a file involves writing), it will require to count the clusters. This can take a few seconds before you receive the response.

1.3 Updating the backup of important settings

From version 1.68 it is possible to make a backup important settings in external flash. When a backup is available, any changes made to the specific settings (e.g. IP settings) will be reverted back to the values from the backup when the device is powered off. For more information on this topic, please see “Supplementary release documentation for software v1.68”.

The only way to update the settings in backup was to modify the settings first and issue the copy command before the device was restarted/powered off.

Sometimes it is required that the IP configuration is changed in the field, for a device with a valid backup in external flash. The person carrying out this change does not always have knowledge of the settings backup or STP. Because of this, in version 1.72 we allow that the backup is updated when a customer changes settings from the device’s menu.

When changes are made to the configuration of the device from the menu and the user exits the setup, a check is done as to whether a valid backup was available in flash. When a valid backup was found, this will be updated. From this moment the device will be using the new IP settings (including any changes made for proxy enable/disable flag).