

# Add or Change DTMF Functions

From "AllStarLink Wiki"

This howto explains how the DTMF function decoder works in app\_rpt and how to add and change DTMF function codes.

**How DTMF Function Decoding Works** The function decoder's job is to collect DTMF digits until a match with a command is found. The function decoder will start collecting digits when the start character (usually \*) is seen, and will continue collecting until the digit time out is reached, or a match is found. The start character is not included in the match. The function decoder does not care what the state of COR or CTCSS are. In other words, if you unkey and re-key in the middle of entering a function, the unkey does not affect the digits collected. Four things will clear the collected digits in the function decoder:

Lack of a match and a digit timer time out. The detection of another start character (usually \*). The recognition and execution of a valid DTMF command The digit buffer is completely filled and no match is found

**Matching Mechanism** A match occurs when the collected digits match a function on the right hand side of an entry in the function stanza. Here's an example:

```
[functions] 1=i1ink,1
```

Above is an example of a function stanza with one entry defined. Since the start character is implied, it is not present on the right hand side of the entry. In this particular case, a \*1 will execute the method 1 of the i1ink function (which is disconnect link). Once a match is detected, any additional digits belong to the function which matched. In this particular case, the digits which follow \*1 will be passed to the disconnect link function where it will attempt to make a match with a connected node.

The first match found will be the one which is chosen. What this means is that you cannot have two DTMF functions same sequence of digits and one having one additional digit because the shorter one will always win and be executed first.

Will not work as intended:

```
[functions] 80=i1ink,1 800=i1ink,2
```

Will work as intended:

```
[functions] 800=i1ink,1 801=i1ink,2
```

**Function Classes And Methods** Let's now devote our attention to the stuff on the right hand side of the equals sign. The right hand side is a list of arguments separated by commas. The first argument is the function class identifier which is a text string. The second argument can be a method number associated with the function class, one or more configuration arguments, or a combination of both. One example of a function class is the i1ink used above. Another example would be the cop function class used for control operator functions. Additionally, instead of or in addition to a method number, the function class itself might accept additional arguments for customizing how it behaves. The autopatchup method is a good example of this:

```
6=autopatchup,noct=1,farenddisconnect=1,dialtime=20000
```

A complete list of the function classes and methods are documented at the top of app\_rpt.c source code file. The app\_rpt source file is located in /usr/src/asterisk/apps on your ACID installation, and in the same directory on a Limey Linux installation after the getsrc script is executed in /usr/src.

Retrieved from "[http://wiki.allstarlink.org/w/index.php?title=Add\\_or\\_Change\\_DTMF\\_Functions&oldid=1223](http://wiki.allstarlink.org/w/index.php?title=Add_or_Change_DTMF_Functions&oldid=1223)"

- This page was last modified on 25 January 2019, at 18:20.