

DTMF Issues

- Overview
 - Sangoma Hardware and DTMF
- DTMF and SIP
- DTMF Troubleshooting Tactics
 - Capturing Audio from the PBX
 - Asterisk - dahdi_monitor recording
 - Freeswitch - ftm trace recording
 - Analysing Recordings using Audacity
 - Digging Deeper - Take a recording from the Echo Canceler

Overview

DTMF (Dual Tone Multi Frequency) is a type of signaling used primarily in voice telephony systems. It features audible tones in the frequency range of the human voice which are typically used when dialing a call (on analog lines) or when operating an IVR menu. There are many other applications for this signaling. For more information, see the wikipedia article on DTMF (http://en.wikipedia.org/wiki/Dual-tone_multi-frequency_signaling).

DTMF tones are generated by combining two tones of different frequency. The combination of two tones of certain frequencies will signify a digit being pressed on the keypad. These frequencies are described in standards which ensure compatibility between different systems. Because DTMF tones are simply audio on the line, they can be heard in any conversation or audio recording on the line. This is referred to as **in-band** signaling because we are using the same channel as the voice conversation to send the signaling tones. Due to the nature of DTMF and inband signaling in general, any noise or distorting of audio on the line will in turn distort the tone, making detection by a remote system impossible. **For DTMF to work reliably, the audio on the line must be clear and of high quality.** It is important to bear this in mind when troubleshooting problems with DTMF.

Sangoma Hardware and DTMF

If your Sangoma card is equipped with a hardware echo canceler, you can use it to detect DTMF tones which come through the card. There are two configuration options available for the Wanpipe driver that relate to DTMF. They are:

- DTMF Detection
- DTMF Removal

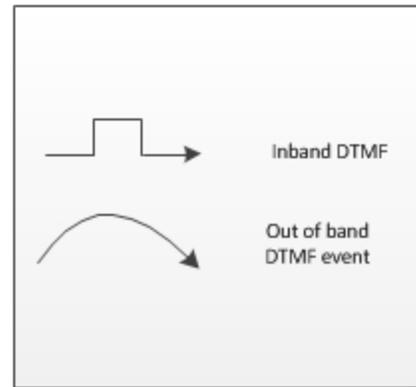
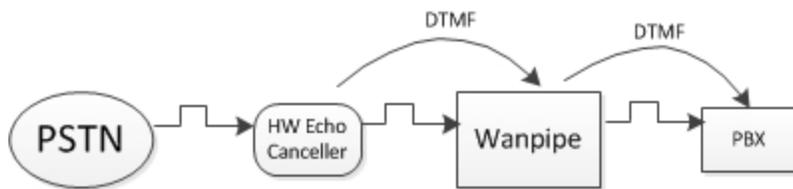
With DTMF Detection enabled, the Sangoma card will use the echo canceler chip to listen for incoming tones. When a tone is detected, it will generate an **out of band DTMF event** which will be sent to the PBX. It is important to note that the actual in-band tone will remain on the line. If you are having problems with digits being doubled (**doubled DTMF**), you might try disabling DTMF detection on your PBX, as it could be detecting both the in-band tone as well as the generated event from the Wanpipe driver. As another option, you could enable DTMF Removal. To activate DTMF detection, edit */etc/wanpipe/wanpipeX.conf* where X is the number of the Wanpipe interface for which you want to enable the feature. Look for the parameter **TDMV_HW_DTMF** and set it to 'YES'.

The DTMF removal feature is used in conjunction with DTMF detection. With this option enabled, DTMF tones detected on the line by the echo canceler will be removed from the audio before it is sent to the PBX. If your PBX is doing DTMF detection as well, you might enable this option to remove tones from the audio so as to avoid doubling of DTMF digits. Please note that in some cases, systems with this option enabled have exhibited a side-effect of removing beeps from the inbound audio (such as the beep from a voicemail system). To activate DTMF removal, edit */etc/wanpipe/wanpipeX.conf* where X is the number of the Wanpipe interface for which you want to enable the feature. Look for the parameter **HWEC_DTMF_REMOVAL** and set it to 'YES'.

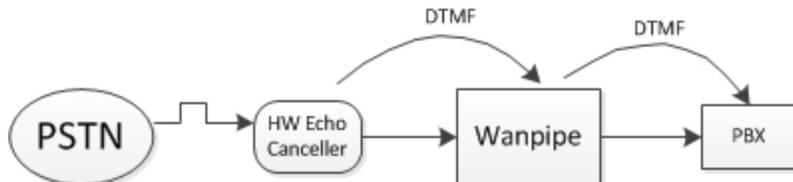
Figure 1 illustrates the concept of DTMF detection and DTMF removal.

Fig. 1

With DTMF Detection Enabled



With DTMF Detection and DTMF Removal Enabled



DTMF and SIP

Because SIP does its signaling over an IP Networks as packet data, it is not necessary to have in-band transmission of DTMF tones. Instead, SIP calls transmit key presses during the call as out of band events. There are three methods for transmitting DTMF in SIP networks:

- RFC 2833
- SIP INFO (RFC 2976)
- Inband

In general, there is very little difference between RFC2833 and INFO. Both send the DTMF as events out of band. For complete details, check the RFCs at <http://www.ietf.org/rfc/>. The third option is to send the tones inband just as a standard telephone would do. Depending on the PBX configuration, any one of these options is viable, but it is important to note that some SIP devices will send the tone as both inband and RFC 2833 or RFC 2976. If you are experiencing **doubled DTMF between the SIP phone and the PBX or the SIP phone and the PSTN**, check your configuration on the SIP phone and the PBX to ensure that you are not using both methods together.

When troubleshooting SIP and DTMF(or and SIP issue), it is often helpful to take a packet capture of a call in progress to examine the flow of events. Wireshark is a very useful tool and has powerful filters and features to dissect a SIP call to see what is going on in the exchange. It is even possible to play back the RTP stream and hear the audio of the call. If there are DTMF issues, listen to the audio stream for inband DTMF and examine the flow of packets for DTMF events. If the audio quality is poor or there is network congestion, this will prevent the normal function of inband DTMF. For full documentation and download for Wireshark, visit <http://www.wireshark.org/>.

DTMF Troubleshooting Tactics

When attempting to diagnose and correct issues with DTMF, the best way to get a picture of what is happening on the line is to take audio recordings from your PBX. This enables you to hear the tones exactly as they are being sent to or received from the Sangoma hardware. If the problem can be heard in the audio to or from the PBX, this will give you a good idea of where the problem is being introduced. This tactic works well for troubleshooting general audio issues as well. Once the audio recording is captured, you can open it with a sound editing program (Audacity for example, which is free and open-source) and listen to the audio or view the waveform.

Capturing Audio from the PBX

The steps to capture audio from the PBX will differ depending on which software is being used. The steps outlined here will deal with Asterisk and Freeswitch, though most PBXs will have some method by which this can be accomplished and could likely be found in the documentation for the software.

Asterisk - dahdi_monitor recording

Before capturing a dahdi recording, it is necessary to know which channel Asterisk is using for the call in question. If you have no calls active on the system and you have configured Asterisk to allocate channels sequentially from low to high, then channel 1 is a good guess, however you may not have control over which channels the provider sends inbound calls on. You can find the channel for an active call by accessing the Asterisk CLI (shell command: asterisk -r) and then giving the command 'core show channels'. This will list channels for active calls, but it's might not be useful to capture audio from an already established call if you are having problems with DTMF dialing. The best way to guarantee the channel you are using is to use the **originate** command to place a call from the Asterisk command line. Take the following steps:

Recording an Outgoing call

1. Open two terminal sessions to the Asterisk system.
2. On the first terminal access the Asterisk CLI (asterisk -r).

Prepare the originate command by typing the following:

```
-> originate dahdi/<channel number>/<number to dial> application playback demo-congrats
```

Make sure to replace <channel number> with a free channel and <number to dial> with a phone number. **Do not hit the return key**, just prepare the command so it's ready to go.

3. Switch to the second terminal and run this command:

```
-> dahdi_monitor <channel number> -v -r streamrx.raw -t streamtx.raw
```

4. Replace <channel number> with the channel you selected in the previous step. Once the command is running, proceed to the next step.
5. Switch back to the first terminal session and hit return to push the call from Asterisk.
6. Once the call is complete, end the dahdi_monitor command. You should have two files in the working directory (streamrx.raw and streamtx.raw). You can then import these files into Audacity for analysis. Steps to do this can be found below.

Recording an incoming call using this technique is a little more tricky, since you have to guess the channel. If it's not critical to capture the first few seconds of the call, you can wait until it is established before find the channel in Asterisk and run the dahdi_monitor command on that channel as described above.

Freeswitch - ftdm trace recording

1. Access the Freeswitch CLI (<freeswitch install path>/bin/fs_cli)
2. Give this command to enable call recording, taking care to fill in the correct parameters based on which channel and span the test call will be on:

```
-> ftdm trace <path> <span_id> <chan_id>
```

Note: The <path> parameter includes both the path for output recordings and the prefix for naming the files. Eg. /root/test will output files named test-in-s1c1 and test-out-s1c1 in /root for example.

3. You can disable tracing when finished with this command:

```
-> ftdm notrace <span_id> <chan_id>
```

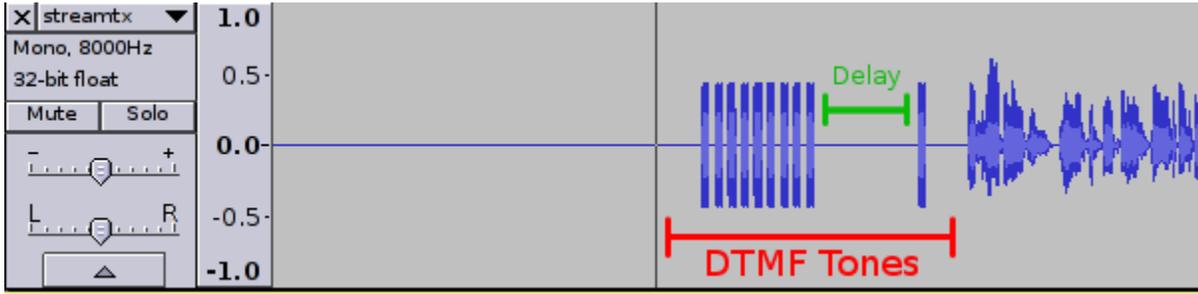
Analysing Recordings using Audacity

Once you have the recordings from the PBX, you can open them in a sound editor such as Audacity. Often, problems with DTMF can be seen visually in the waveform. The files output from Asterisk and Freeswitch are raw audio files. They don't have a container which specifies the codecs and formats, so you'll have to tell Audacity how to play them. Open up Audacity and select a stream to import by selecting **File->Import->Raw Data...** Browse to the recording files you took from the PBX. You'll have to import the Tx and Rx streams one at a time. When you select the file, Audacity will ask you for some information about the audio stream. Use the following settings:

Parameter	Value
Encoding	U-Law or A-Law (depending on your config)
Byte-Order	No Endianness
Channels	1 Channel (Mono)

Start Offset	0
Amount to import	100%
Sample Rate	8000 Hz

Pictured here is an example of what the imported audio should look like in Audacity. In this example, we can see that a 10 digit number is being dialed, but there is a long delay before the last digit is being sent. Because this delay is present in the audio leaving the PBX, the problem must be solved in the PBX config. In this case, the PBX is asterisk and it was necessary to disable echotraining in /etc/asterisk/chan_dahdi.conf.



Using this technique of taking recordings and analysing, you can isolate the cause of DTMF issues so you can focus troubleshooting efforts in the right place. Here are some common issues we see with DTMF.

Incoming calls from PSTN:

Observations	Conclusion
DTMF tones are correct and present (but aren't heard by SIP devices)	Problem exists in the PBX configuration
DTMF tones are garbled or missing	Problem exists in the Wanpipe driver, Sangoma hardware, or the PSTN

Outgoing calls to the PSTN

Observation	Conclusion
DTMF tones are missing or garbled	Problem exists in the PBX. Tones are mangled or missing before being sent to the line. Check PBX configuration and documentation.
(Asterisk) There is a long delay before the last tone is sent in a dialed number.	Turn off "echotraining" in /etc/chan_dahdi.conf by setting it to 'no' or commenting it out.
DTMF tones are present, but calls won't connect	Check that the PBX isn't dialing digits too quickly after picking up the line. Add a 'wait' in the dialplan before dialing number. If this fails, problem exists in configuration, driver or hardware.

Digging Deeper - Take a recording from the Echo Canceler

If through previous troubleshooting steps you are able to determine that the DTMF tones are leaving the PBX intact and working, OR if you notice that all incoming DTMF tones are missing or mangled in some way, the next thing to check is the Hardware Echo Canceler Chip (if your card is equipped with one). If your card does not have an echo canceler chip, then there is nothing between the PSTN and your PBX that would manipulate the audio in any way, therefore the problem is caused by the physical line from the telco, or elsewhere in the PSTN network.

As a first step, try disabling your hardware echo canceler and make some test calls to see if the handling of DTMF is improved. To do this, run the following command:

```
-> wan_ec_client wanpipe1 disable
```

This command will disable echo cancellation for the wanpipe1 device. If you have multiple wanpipe interfaces in your system you might need to repeat this command for each span. To reverse the changes, give the command with "enable" or restart wanrouter.

If you note that turning off the echo canceler improves the handling of DTMF, the next step is to take a recording from the echo canceler chip. To do this, see our [instructions](#).