

# Performance of Various Codecs Related to Jitter Buffer Variation in VoIP Using SIP

**Iwan Handoyo Putro**

Electrical Engineering Department, Faculty of Industrial Technology  
Petra Christian University  
Siwalankerto 121-131, Surabaya 60256  
Email: iwanhp@petra.ac.id

## ABSTRACT

Briefly speaking, there are two popular Voice over Internet Protocol (VoIP) standards, H.323 and Session Initiation Protocol (SIP). The first standard was designed by ITU and has become the basis for the widespread implementation of VoIP systems although it was not specifically designed for it. The second standard, SIP, was proposed by IETF and it is designed to connect, communicate and exchange data with the internet applications. In order to deliver voice conversation through packet-switching networks, codecs (coder-decoder) should be implemented to compress and later decompress those packets. In this paper, some of compression algorithms will be compared and analyzed based on its performances in SIP based VoIP system. The codecs that was used in this experiment are SJ Lab GSM 6.10, SJ Lab iLBC-30ms, SJ Lab iLBC-20ms, Microsoft CCITT G.711 A-law and Microsoft CCITT G.711 u-law. These codecs are tested in terms of its ability to deal with jitter buffer variations. The result shows that SJ Lab iLBC-20ms gives the best performance in terms of jitter buffer variation on LAN environment while SJ Lab GSM 6.10 shows the highest performance on wireless networks testing.

**Keywords:** VoIP, codecs, SIP, jitter buffer

## INTRODUCTION

The history of delivering voice over computer networks has started at 1995 when a small company, Vocaltec, released their first internet phone software called Internet Phone. In order to work, this software needs sound card, microphone, speakers and uses H.323 protocol[1]. At that time, the voice quality of VoIP service was poor as the result of limited bandwidth and modems technology.

The number of VoIP users increased significantly in the past 3 years with the implementation of broadband internet access. As shown on figure 1, it is predicted that the number of VoIP users worldwide increase significantly from about 480,000 users in 2006 to almost 70.6 million in 2011[2].

Some packet-switched network operators are interested to implement VoIP service as voice transmission through IP networks because it would give additional income without adding too much implementation cost.

Building a VoIP network on internal network would require almost no cost if the IP network has already existed.

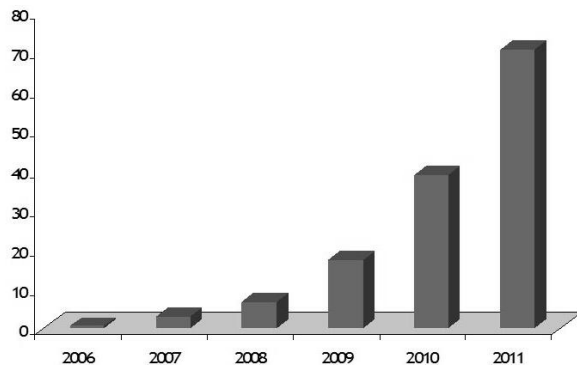


Figure 1. Growth in worldwide VoIP users[2]

Figure 2 shows that when the existing LAN has been built, the minimum requirement is just installing a pair of soft phone which basically behave like a traditional PSTN phone, i.e. dialing, calling and terminating the conversation.

However, because of the increasing demand of broader coverage of communication and the mobility, some communication equipment such as router, phone adapter and a reasonably fast internet connection should be added.

Basically, delivering voice through IP networks requires some processes at both the caller and the callee side. At the caller side, the voice is converted from analog into digital signal using the codecs. The digital packet then would be put on the IP packets together with its source and destination address before

**Note:** Discussion of this paper must be submitted before December 1<sup>st</sup>, 2008. The proper discussion will be published in Electrical Engineering Journal volume 9, number 1, March 2009.

sending them over the networks. On the other hand, at the callee site, the reversion process should be done so that the callee could hear the original voice[4].

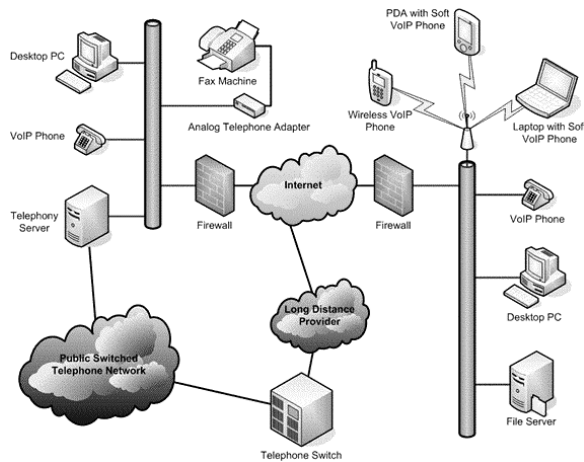


Figure 2. The example of VoIP network [3]

Recently, there are two popular VoIP standards that widely used. They are H.323 and SIP. Both of those standards are responsible for establishing call, maintaining connection and translating IP addresses as well as telephone numbers, bandwidth management, authentications and registration [5]. However, SIP is considered because it is easier to understand, extend and implement than H.323 [6] and furthermore, it is similar to Internet/web in terms of syntax and architecture [7].

### SESSION INITIATION PROTOCOL (SIP)

The Session Initiation Protocol (SIP) is protocol standard that was set up by IETF and has aims to deliver voice over internet protocol in such simpler and modular way[8]. It is widely used to set up two points for multimedia communication sessions such as voice and video calls over the Internet[5].

SIP defines the telephone number like a URLs, therefore, it can be embedded into web pages. On another word, SIP could interwork with the internet application well[8].

Similar to H.323 protocol, SIP could be used for establishing, maintaining as well as terminating 3 kinds of sessions, such as two-party, multiparty and multicast sessions. Those sessions could consist of audio, data and video.

Based on OSI model, the SIP protocol is placed in the session layer part, while on the TCP/IP model, it is positioned at the application layer [9] as a result of their design which is independent from the lower layer. SIP could easily run on some transport technology, such as TCP, UDP and SCTP. The way SIP establish its connection could be seen on figure 3.

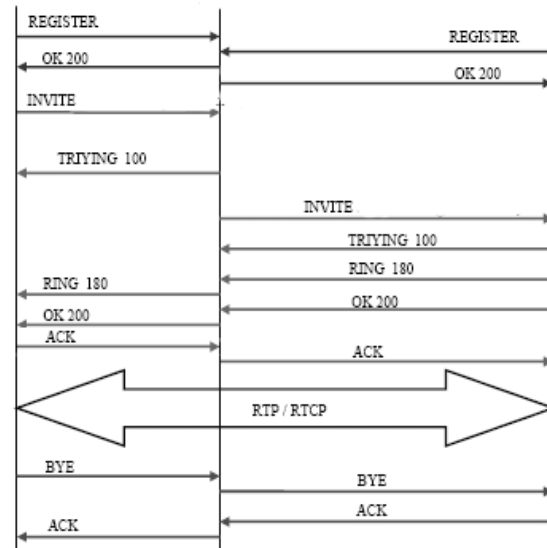


Figure 3. SIP call flow diagram [10]

### CODEC

Codecs play an important role in VoIP system. Because human voice is analog signal, it should be changed into digital signal before entering the IP networks. On the other side, that digital form should be converted into analog signal in order to get the original message.

Next, it has a responsibility to compress voice so that could be delivered via IP network. It means the packet size should be reduced so it requires minimum bandwidth.

Two important stages in digitization process are sampling and quantization. Sampling is the process of converting a signal into numeric sequence[11]. Quantization process is the process of approximating a continuous range of values by a set of discrete representation[12]. Table 1 shows the list of codecs that would be used in this experiment.

Table 1. List of codec comparison[7]

Codec	Sampling rate	Bandwidth (kbps)	License	Bit Rate	MOS
SL Lab GSM 6.10	8 kHz	13.2	Open Source	13 kbps	-
SJ Lab iLBC-30ms	8 kHz	15	Royalty Free	33 kbps	4.14
SJ Lab iLBC-20ms	8 kHz	15	Royalty Free	15.2 kbps	4.14
Microsoft CCITT G.711 a-law	8 kHz	64	Open Source	64 kbps	4.45
Microsoft CCITT G.711 u-law	8 kHz	64	Open Source	64 kbps	4.45

**TESTING AND RESULT**

The purpose of this experiment is to investigate which codecs would give better performance on VoIP using SIP. The experiments were done in variation of jitter buffer. The result of this experiment could be used for further research to obtain optimum size of jitter buffer when delivering voice through IP Networks. The research tool that was used in this research is relatively simple and most of the used software is freeware.

In VoIP, the jitter buffer terminology refers to a shared data area where voice packets would be collected and stored, before it is sent to the next voice processing level[13]. Here, the packets should be stored first in order to minimize delay variations which could be caused by route changes, timing drift and network congestion.

In terms of jitter buffer size, two problems could be occurred. First, when a jitter buffer is configured too small then an excessive number of packets may be discarded. On the other hand, if more space is given then delay would be longer[14].

There are two common types of jitter buffer, i.e. hardware based and software based. The first type of jitter buffer is configured by the manufacturer while the second one is a dynamic jitter buffer and therefore it can be easily configured to adapt with the variation of the IP networks [14].

Figure 4 shows the network configuration used in this research. Here, two PC are connected through LAN. One of them has PCI wireless adapter and is connected to other PC that equipped with wireless card adapter through wireless AdHoc configuration.

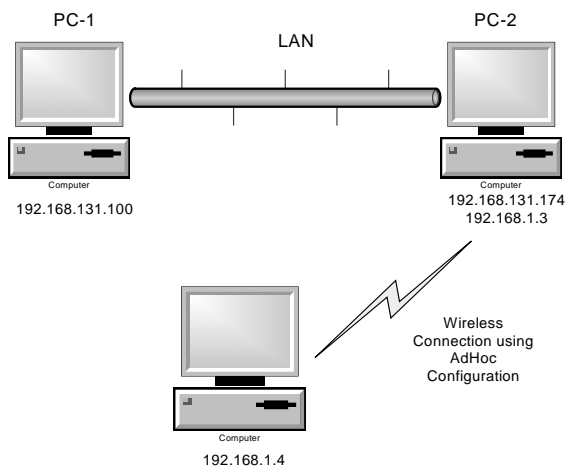


Figure 4. Network configuration that is used in the testing period

On the PC, SPhone is installed and would be used as software phone[15]. The VoIP traffic is captured by using Wireshark[16] to get the percentage of the packet drop according to the jitter buffer variations. The testing was done in LAN and Wireless Network environment and the result could be seen in table 2 and 3 while the graphic representation of the result is presented in figure 5 and 6.

**Results on LAN Testing**

Table 2. Codecs performance in terms of dropped packets as a result of jitter buffer variation on LAN testing

Jitter Buffer (mS)	Codecs				
	SJLab GSM 6.10 (%)	SJLab iLBC-30ms (%)	SJLab iLBC-20ms (%)	Microsoft CCITT G.711 A-law (%)	Microsoft CCITT G.711 u-law (%)
	1	80,9	96,2	97,5	99,3
2	77,1	89,1	93,2	98,1	95,1
3	77,1	80,2	92,4	94,6	93,3
4	77	79,1	91,4	91,1	88,6
5	77	77,6	90,9	85	82,2
6	77	77	90,8	83,6	75,3
7	76,7	77	87,9	80,3	72,2
8	68,5	75,9	53,4	70,4	65,3
9	60,1	72,3	49,2	65,2	59,4
10	55,1	71,2	30	55,1	48,5
11	54	71,2	26,3	48,5	41,9
12	51	70,5	17,6	36,7	31
13	46,3	65,4	12,1	29,2	24,1
14	41,5	57,5	27,1	24,3	20,1
15	30,5	47,9	10,7	19,6	17
16	21,2	43,3	1,8	14,8	18
17	16,4	41	1,4	10,1	11,2
18	15,2	40,2	1	6,2	7,7
19	16,7	39,6	3,1	3,1	14,6
20	22,5	37,5	1,2	11,8	6,5
21	6,5	36,5	0	9,3	4,3
22	2,2	36,1	0	7	2,6
23	0,3	33,3	0	4,6	1,5
24	13,9	19,1	0	2,8	0,7
25	11,3	8,3	0	1,4	0,2
26	9,7	6,5	0	0,7	11,4
27	3,6	2	0	0,2	9,8
28	0,4	0,8	0	0,1	9
29	0	1,4	0	0,1	8,6
30	0	7	0	0,1	6,7

Jitter Buffer (mS)	Codecs				
	SJ Lab GSM 6.10 (%)	SJ Lab iLBC-30ms (%)	SJ Lab iLBC-20ms (%)	Microsoft CCITT G.711 A-law (%)	Microsoft CCITT G.711 u-law (%)
	31	0	6,9	0	0,1
32	0	1,9	0	0,1	2,2
33	0	1,2	0	0,1	1,1
34	0	1,2	0	0,1	0,5
35	0	10,8	0	0,1	0,5
36	0	8,4	0	0,1	2,3
37	0	6,3	0	0,1	1,2
38	0	6,3	0	0,1	0,8
39	0	10,7	0	0,1	1,5
40	0	6,5	0	0,1	2,1
41	0	2,9	0	0,1	18,4
42	0	0,3	0	0,1	15,9
43	0	3,6	0	0,1	12,9
44	0	3,1	0	0,1	10,5
45	0	0,2	0	0,1	8,6
46	0	0	0	0,1	6
47	0	0	0	0,1	4,1
48	0	0	0	0,1	2,5
49	0	0	0	0,1	1,2
50	0	0	0	0,1	0,4

Table 2 shows that SJ Lab iLBC-20ms gives the best performance in terms of packet dropped percentage in LAN testing. This Codec needs only 21 mS size of jitter buffer in order to deliver 0% packet dropped. The worst performance is shown by both Microsoft CCITT G.711 A-law and u-law. Those codecs could not achieve zero packets dropped free although they use 50 mS jitter buffer size.

This testing result also proofs that when the amount of jitter buffer is too small, a great number of packets maybe discarded. On the other hand, allocating too much jitter buffer could add more delay which lead to communication problem [17].

Figure 5 shows that the increasing size of jitter buffer is not linear to the decreasing number of packet dropped percentage. Here, it is easy to conclude which codec could give a good performance in terms of jitter buffer variation in wired-network. By looking at the codec that achieve 0% packet dropped using minimum size of jitter buffer, the best performance is resulted by SJ Lab iLBC-20ms. This good performance of iLBC-20ms codec could possibly happen because of its design which is suitable for narrowband speech and less bandwidth needed [18].

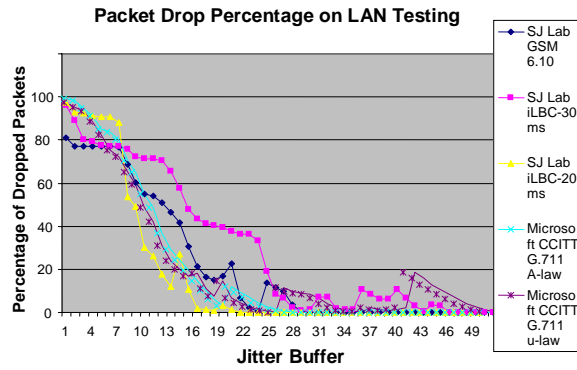


Figure 5. Packet drop percentage on LAN testing

### Results on Wireless Testing

Table 3. Codecs performance in terms of dropped packets as a result of jitter buffer variation on wireless network testing

Jitter Buffer (mS)	Codecs				
	SJ Lab GSM 6.10 (%)	SJ Lab iLBC-30ms (%)	SJ Lab iLBC-20ms (%)	Microsoft CCITT G.711 A-law (%)	Microsoft CCITT G.711 u-law (%)
	1	96,8	91,3	97,8	92,6
2	92,8	87,2	94,4	89,7	92,7
3	91,9	76,2	93,2	86,3	87
4	90,7	73,5	92,4	78,2	75,9
5	90	66,4	91,9	67	65,3
6	89,9	64,9	91,8	59	52,5
7	87	64,6	88,3	47,6	46,5
8	52,7	64,1	55,6	40,4	38,3
9	50,5	61,3	52,1	29	31,9
10	30	59,6	30,3	21,6	26,4
11	22,8	58,9	24,1	14	15,5
12	13,1	50,4	17,6	29,3	8,7
13	12,2	40,1	18,8	25	8,8
14	26,3	42,3	26,3	20	9,6
15	16,8	42,1	10,5	15	33,2
16	1,2	37,7	7,1	10	8,6
17	1,2	29,3	1,8	5,6	7,3
18	1	17	0,8	3	6,3
19	1,1	15,9	5,2	1,6	6,2
20	0,6	4,8	8,7	3,9	6,1
21	0,1	4,8	2,4	1,7	5
22	0	4,6	2,4	0,9	2,7
23	0	4,3	2,4	0,8	5
24	0	11,8	2,4	0,8	2,6
25	0	3,1	2,4	0,8	0,8
26	0	3,1	1,9	0,8	0,3
27	0	2,4	1	0,5	0,2

Jitter Buffer (mS)	Codecs				
	SJ Lab			Microsoft	Microsoft
	GSM 6.10 (%)	iLBC-30ms (%)	iLBC-20ms (%)	CCITT G.711 A-law (%)	CCITT G.711 u-law (%)
28	0	2,2	2,4	0,5	0,5
29	0	2,3	2,3	0,4	0,5
30	0	3,8	2	3,7	0,7
31	0	2,9	0	31,5	0,7
32	0	1	0	26,5	27,2
33	0	0,7	0	22,3	23,7
34	0	0	0	17,5	19,1
35	0	0	0	12,6	14,1
36	0	0	0	7,6	9,4
37	0	0	0	4,1	5,7
38	0	0	0	2,2	2,6
39	0	0	0	0,8	1,1
40	0	0	0	0,1	0,3
41	0	0	0	0	0
42	0	0	0	0	0
43	0	0	0	0	0
44	0	0	0	0	0
45	0	0	0	0	0
46	0	0	0	0	0
47	0	0	0	0	0
48	0	0	0	0	0
49	0	0	0	0	0
50	0	0	0	0	0

While in the wired testing, SJ Lab iLBC-20ms could perform best packet dropped handling with minimum size of jitter buffer, on the wireless testing, the SJ Lab GSM 6.10 could achieve zero packets dropped with minimum size of jitter buffer 22 mS.

As a rule of thumb, a typical size of jitter buffer is about 30–50 milliseconds [17]. From the Table 3, it is shown that the ability of each codec to deliver zero packets dropped is varied. However, all of the codecs could achieve zero packets dropped in the jitter buffer variation range of 30–50 mS.

It has been investigated before [19, 20] that the toleration of packet lost in VoIP communications is no more than 1%. From Figure 6 two codecs, Microsoft CCITT G.711 A-law and u-law, give an interesting result. When the jitter buffer size exceeds of 30 and 32mS respectively, the percentage of packet dropped increases significantly and reaches its maximum size when the jitter size is given at 33mS. After that the number of packet dropped decreases gradually and could deliver zero percentage at 41mS

jitter buffer size. Regarding to this result, further investigation should be done in order to get an explanation whether it is a specific codec related to problem because of this anomaly does not appear on other three codecs.

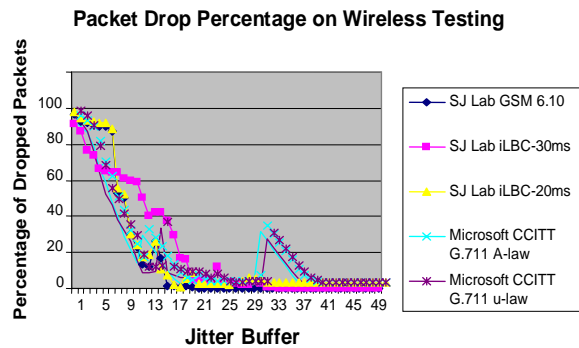


Figure 6. Packet drop percentage on wireless testing

### CONCLUSIONS

In this paper, various codecs has been tested in both wired and wireless environment on isolated network. From the experiment result, it is shown that SJ Lab iLBC-20ms gives the best performance in terms of jitter buffer variation on LAN environment. And SJ Lab GSM 6.10 shows the highest performance on wireless networks testing.

Other fact has been found that the increasing number of jitter buffer size is not linear to the decreasing of packet drop percentage. It happened to Microsoft CCITT G.711 u-law codecs on LAN testing and both of Microsoft CCITT G.711 (a-law and u-law) on wireless networks testing.

On the future works, the result on non-isolated networks (i.e. via proxy) will be investigated and compare with the previous work.

### ACKNOWLEDGMENT

I would like to give my appreciation to Mr. Fredy Matulesy who has given a helpful assistant during the testing period and people at Telematics Laboratory, for giving a chance to use their networks and some supporting equipments.

### REFERENCES

- [1] *History of VoIP*, 11<sup>th</sup> March 2009 <<http://www.whichvoip.com/blog/history-of-voip> accessed>
- [2] *Consumer Mobile VOIP: Industries Collide* <<http://www.onworld.com/mobilevoip/images/cellvopusers.jpg>>, 10<sup>th</sup> March 2009.

- [3] *Introduction to VoIP Technology*, 10<sup>th</sup> March 2009 <<http://www.pcmec.com/article/introduction-to-voip-technology>>
- [4] Wikipedia, *Voice over Internet Protocol*, 11<sup>th</sup> March 2009. <[http://en.wikipedia.org/wiki/Voice\\_over\\_Internet\\_Protocol](http://en.wikipedia.org/wiki/Voice_over_Internet_Protocol)>
- [5] *H.323 versus SIP: A Comparison*, 11<sup>th</sup> March 2009 <[http://www.packetizer.com/ipmc/h323\\_vs\\_sip](http://www.packetizer.com/ipmc/h323_vs_sip)>
- [6] *Understanding SIP*, 12<sup>th</sup> March 2009 <[http://www.sipcenter.com/sip.nsf/html/WEBB5YNVK8/\\$FILE/Ubiquity\\_SIP\\_Overview.pdf](http://www.sipcenter.com/sip.nsf/html/WEBB5YNVK8/$FILE/Ubiquity_SIP_Overview.pdf)>
- [7] Iwan Handoyo Putro, Indar Sugiarto. "Performance Analysis of Different Codecs in VoIP using H.323", *Proceedings of ISSTEC2009*. Yogyakarta, 24<sup>th</sup> January 2009.
- [8] Tanenbaum, A., *Computer Networks*, 4<sup>th</sup> ed., New Jersey: Prentice Hall, 2003.
- [9] *Session Initiation Protocol*, 12<sup>th</sup> March 2009 <[http://en.wikipedia.org/wiki/Session\\_Initiation\\_Protocol](http://en.wikipedia.org/wiki/Session_Initiation_Protocol)>
- [10] *SIP Communication Example* <[http://www.voipforo.com/en/SIP/SIP\\_example.php](http://www.voipforo.com/en/SIP/SIP_example.php)>
- [11] Wikipedia, *Nyquist–Shannon sampling theorem*, 13<sup>th</sup> March 2009 <[http://en.wikipedia.org/wiki/Nyquist%E2%80%93Shannon\\_sampling\\_theorem](http://en.wikipedia.org/wiki/Nyquist%E2%80%93Shannon_sampling_theorem)>
- [12] Wikipedia, *Quantization (sound processing)*, 13<sup>th</sup> March 2009 <[http://en.wikipedia.org/wiki/Quantization\\_%28sound\\_processing%29](http://en.wikipedia.org/wiki/Quantization_%28sound_processing%29)>
- [13] *Jitter Buffer* <[http://searchunifiedcommunications.techtarget.com/sDefinition/0,,sid186\\_gci906844,00.html](http://searchunifiedcommunications.techtarget.com/sDefinition/0,,sid186_gci906844,00.html)>, 12<sup>th</sup> March 2009.
- [14] *Problem: Jitter Buffer*, 12<sup>th</sup> March 2009 <<http://www.voiptroubleshooter.com/problems/jitterbuffer.html>>
- [15] *SJLabs Official Website* <<http://www.sjllabs.com/sjp.html>>
- [16] *Wireshark Official Website* <<http://www.wireshark.org/>>
- [17] *Problem: Jitter Buffer Configuration*, 11<sup>th</sup> March 2009 <<http://www.voiptroubleshooter.com/problems/jbconfig.html>>
- [18] *What is iLBC?*, 11<sup>th</sup> March 2009 <<http://www.ilbcfreeware.org/>>
- [19] Markopoulou A.P., Tobagi F.A, Karam M.J., "Assessing the Quality of Voice Communications Over IP Backbones," *IEEE/ACM Transactions on Networking*, vol 11, no 5, October 2003, pp. 747-760
- [20] James J.H., Chen B., Garrison L., "Implementing of VoIP: A Voice Transmission Performance Progress Report", *IEEE Communication Magazine*, July 2004, pp. 36-41
- [21] Toncar, V., *VoIP Basics: Overview of Audio Codecs*, 11<sup>th</sup> March 2009 <[http://toncar.cz/Tutorials/VoIP/VoIP\\_Basics\\_Overview\\_of\\_Audio\\_Codecs.html](http://toncar.cz/Tutorials/VoIP/VoIP_Basics_Overview_of_Audio_Codecs.html)>