Introduction to Telephony

PacNOG 3 VoIP Workshop
June 2007, Cook Islands

Jonny Martin
jonny@jonnynet.net

Analogue Telephony

- Where it all started!

- PSTN allows connection between any two endpoints

- Human speech typically in the range 250 - 3,000Hz

  - Humans can hear in the region of 20 - 20,000Hz

- PSTN analogue channel originally designed to carry 300 - 3,500Hz

- Most analogue lines delivered via copper from the local exchange (or CO, Central Office)

  - Average line in NZ ~3Km. Longest lines >7Km

Analogue Telephony

- Even in the day and age of VoIP, this is still important!

  - Analogue telephone adapters (ATAs)

  - Fax - it just won't go away :)

  - Echo

  - Voice and sound is most definitely analogue

    - First and last conversions in a VoIP call

The Analogue Telephone

- Analogue telephones connect to a copper pair

  - A two wire circuit

- Analogue telephones are comprised of five major parts:

  - Ringer

  - Dial Pad

  - Hybrid

  - Hook switch

  - Handset

Ringer

- The exchange provides DC (~48vDC) to power the phone

  - Exchange = big centralised UPS

- Exchange provides a burst of AC (~80vAC) to ring the phone's bell

  - Originally a mechanical bell, these days an electronic buzzer

- These days phone have a Ringer Equivalence Number (REN)

  - Exchange can power up to a sum of 5 RENs

  - Phones these days typically < 0.5 REN

  - ATAs have same limitation

Dial Pad

- Telephones need to signal back to the exchange

- Originally done with a rotary dialler making and breaking the copper loop

  - Pulse Dial, still typically supported by exchanges and some VoIP kit

- All done with audio tones now

  - Dual Tone Multi Frequency (DTMF)

  - Telephone handsets a matrix of switches

  - One tone per column, one per row

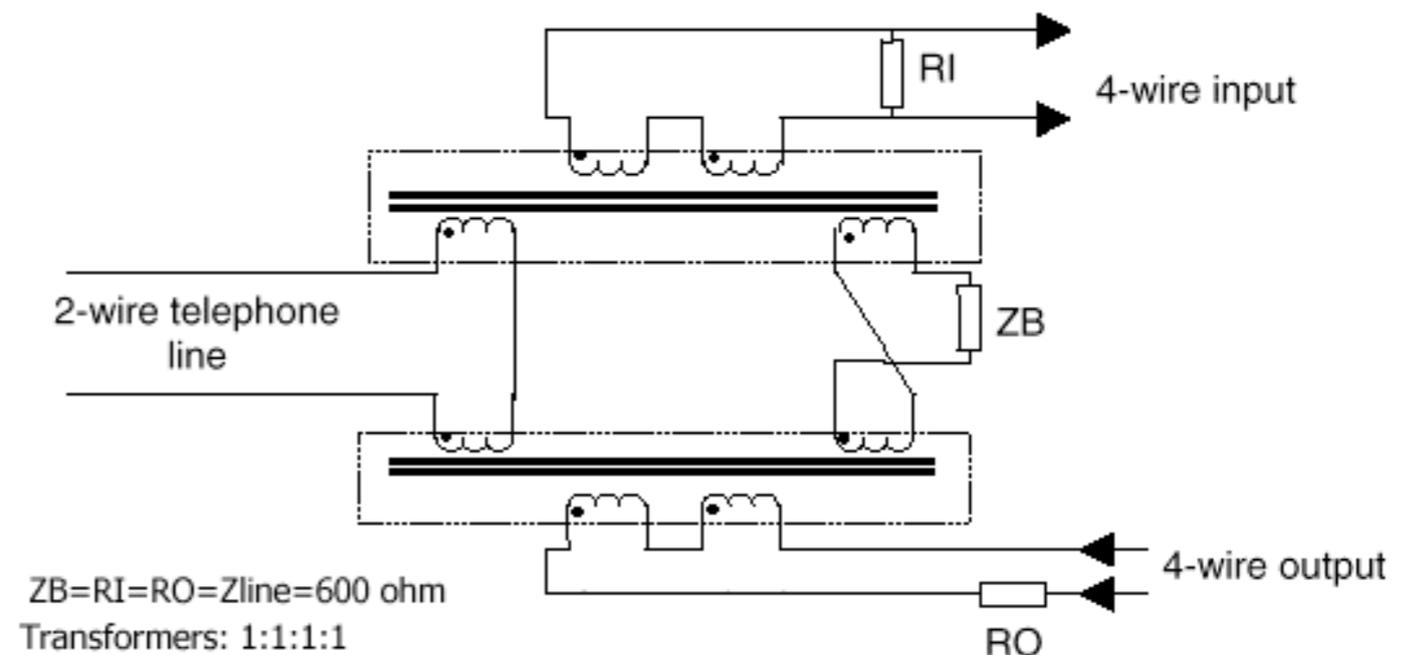 - Each switch generates two tones, hence Dual Tone

DTMF Tones

	1209 Hz	1336 Hz	1477 Hz	1633 Hz
697 Hz	1	2	3	A
770 Hz	4	5	6	B
852 Hz	7	8	9	C
941 Hz	*	0	#	D

Hybrid Network

- The heart of an analogue telephone

- The transformer that couples two signals onto one line

  - Send (Tx) and receive (Rx)

- Creates sidetone ('good echo')

  - Allow speaker to hear himself

- Creates echo unless perfectly balanced

RI

4-wire input

2-wire telephone line

ZB

ZB=RI=RO=Zline=600 ohm
Transformers: 1:1:1:1

4-wire output

RO

Hook Switch

- Telephone uses it to signal state to the exchange

    - On Hook, closes the copper loop

        - Phone idles, waiting for incoming ring

    - Off Hook, breaks the copper loop

        - Requests dial tone from the exchange, and then allows audio to pass

- Also used to signal 'advanced' features, e.g. call waiting

   - Hook Flash - a timed closure of the hook switch, typically ~300ms

Tip and Ring

- Telephony world often refers to 'Tip' and 'Ring'

- Historical term from the days when exchanges were literally switchboards

- Operator manually patched lines together

- Tip (red) = +ve polarity (0v)

- Ring (green) = -ve polarity

  - -48v on hook, -7v off hook

Ring   Tip
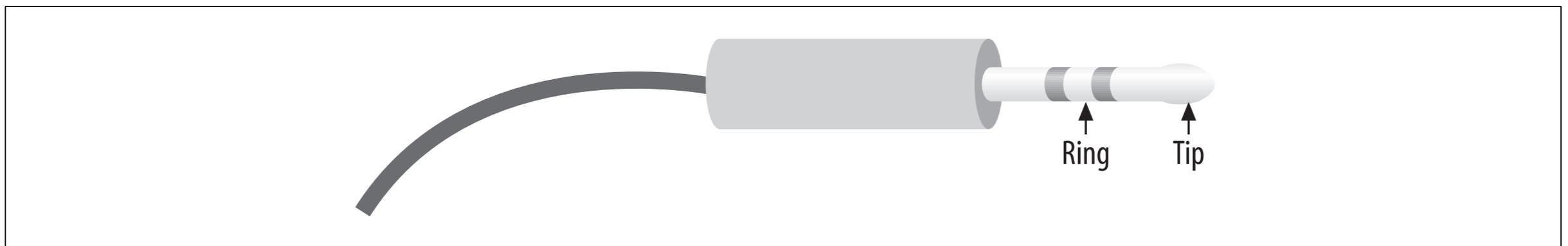
Telephone and Line Impedance

- Impedance = technical way of saying resistance

  - Varies with both frequency and phase

- American telephone impedance is 600 ohms

  - Approximation of the impedance of 0.4mm twisted copper pair at voice frequencies

- British (and NZ) telephone impedance is complex (in the resistive sense of the word), called BT3

  - 370 ohms in series with (620 ohms in parallel 310nF)

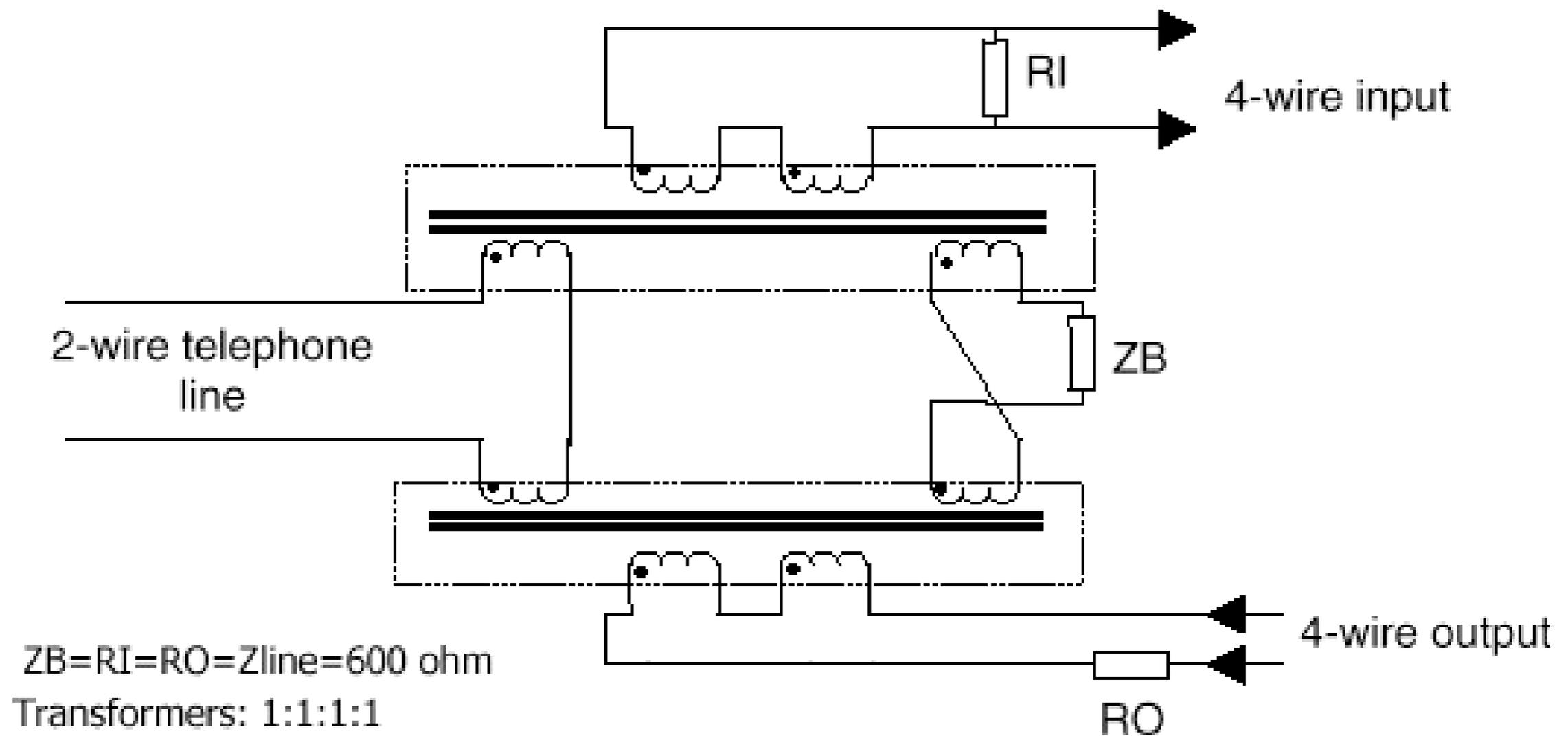 - Attempt to better match line impedance
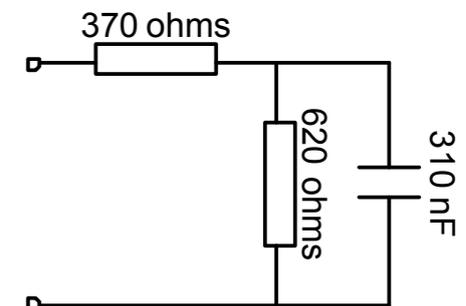
Echo

- VoIP does not cause Echo!

  - Hybrids cause echo

  - Echo becomes apparent as latency increases

  - VoIP creates higher latency than circuit switched circuits

- Hybrids must be balanced to the line to effect maximum power transfer and minimal signal reflection

  - Reflection back down the line = echo

  - Reflection back towards the handset = sidetone

Echo - Telephone Hybrid

RI

4-wire input

2-wire telephone line

ZB

4-wire output

RO

ZB=RI=RO=Zline=600 ohm
Transformers: 1:1:1:1

Echo

- Sidetone is used to let the user know that the phone 'is working'

    - It's somewhat unnatural to not her oneself

    - Too much sidetone and you can only hear yourself

    - Too little and it appears the line is dead

- Echo is present on most lines

    - When latency is low (< 20ms or so) the far end perceives it as sidetone

Acoustic Echo

- Caused by the output of the handset's speaker entering the microphone

    - Due to the speaker volume being too loud or microphone sensitivity too loud

        - Very bad with softphones when not using a headset

    - Or flimsy handset construction (acoustic coupling through the handset itself)

    - The telephone handset design hasn't changed much over the years as it is a very good one!

- Indistinguishable to the far end from echo caused by the local hybrid

Reducing Echo

- There are only four ways to reduce echo

  - Remove the two wire (analogue) portion of the call

  - Balance the analogue portion of the call better

    - Hard to do even if you do have access to the endpoint(s)

  - Reduce the latency

    - Often impossible, e.g. long distance calls

  - Cancel the echo

Echo Cancellers

- Measure signal on the line, predict the echo, and create a signal to cancel it

- Echo cancellers are configured for a 'tail' length - the maximum latency of an echo which it can possibly cancel

- Takes time to converge to an echo cancelled state, dependant on the tail length of the canceller

- Echo cancellers aren't perfect, so best to prevent echo in the first place

- Popular misconception that software based echo cancellation is bad.

  - Hardware echo cancellers have very good, often patented algorithms

  - No really good open source software implementations (yet...)

  - Software echo cancellation is not bad - if you have a good algorithm!

Digital Telephony

- Telephony moved digital for the same reason everything else did

- Voice turned to a digital signal using Pulse Code Modulation (PCM)

    - Sample signal in time

- Two important factors:

    - Number of samples per second (highest frequency is half of the sample rate - Nyquist's Theorem)

    - Number of bits used to encode signal

- Tradeoff between quality and bandwidth - standard is 8bits at 8kHz sampling

Digital Telephony

- Standard voice channel (timeslot, or DS0) is 64kbit/s

- Most common codec is G711, a companding codec

  - Two types, ulaw (US) and alaw (Europe)

- Majority of telephone conversation is 'quiet'

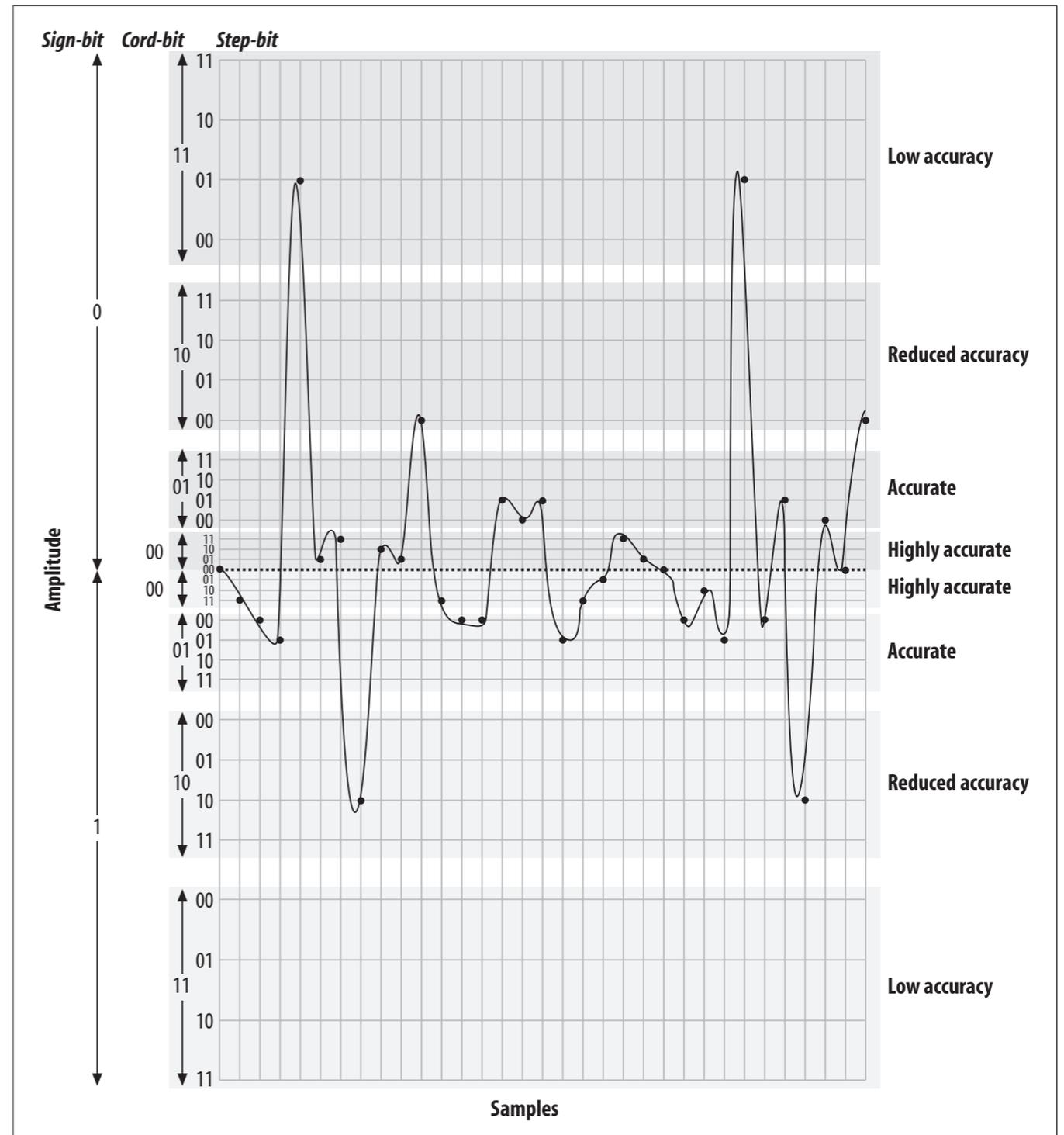- More bits are allocated to quiet signals to improve overall quality

Figure 7-12. Quantized and companded at 5-bit resolution from Asterisk, The Future of Telephony

PSTN Circuits

- Analog line

- ISDN

    - Basic rate, two voice 64kbit/s voice channels + 16kbit/s data channel -> 144kb/s

    - Primary rate

        - US - T1, 24 64kbit/s voice channels -> 1.544Mb/s line rate

        - Europe - E1, 30 64kbit/s voice channels -> 2.048Mb/s line rate

- Proprietary circuits between key phones and PBXs - not covered here

VoIP

- Natural progression from digital telephony

  - Circuit switched --> packet switched

  - Still a need to sample and encode signals

- Many different codecs in the VoIP world

- Many different signalling protocols

Codecs

Codec	Payload Bitrate
G.711	64 kbit/s
G.726	16,24, or 32 kbit/s
G.723.1	5.3 or 6.3 kbit/s
G.729	8 kbit/s
GSM	13 kbit/s
iLBC	13.3 or 15.2 kbit/s
Speex	2.15 to 22.4 kbit/s

- G711 gives highest quality

- Some wide bandwidth codecs supported now

- GSM very popular - good CPU time vs. bandwidth tradeoff

- Speex well suited to changing network conditions

Signalling Protocols

- Signalling protocols needed to allow endpoints and intermediary devices to set up calls

- Common VoIP signalling protocols:

  - H.323

  - MGCP (Media Gateway Control Protocol)

  - Skinny / SCCP (Skinny Client Control Protocol)

  - IAX (Inter Asterisk eXchange)

  - SIP (Session Initiation Protocol)

H.323

- 10 year old ITU protocol developed to carry multimedia traffic across an IP network

  - Actually a suite of protocols, the signalling component being H.245

  - Originally designed for video conferencing

- Quickly became de-facto standard for VoIP - and is still used today in many large carrier environments

- Relatively secure and bug free due to its maturity

- Does not work well with NAT at all

- Has all but disappeared in end stations over the past few years

MGCP

- IETF standard, RFC 3345 (obsoletes RFC2705

- Still widely deployed

  - Slowly being displaced by SIP

- Being a gateway protocol, has very good gateway features useful for a carrier environment

- Some end phone support for MGCP but never big

Skinny / SCCP

- Cisco Proprietary protocol

  - Originally developed by Selius Systems in the mid 1990's

  - Cisco bought them and entered the telephony market :)

- Cisco CallManager based on Skinny, though finally moving to the more standard SIP

- Called Skinny as phones are 'dumb'.

  - SCCP phone events: button X pressed, turn on lamp X, turn off lamp X

IAX

- Developed by Digium, creators of Asterisk

  - Apparently it's pronounced "eeks". I still say "eye-aye-ex"

- Primarily designed to connect Asterisk servers together

  - Has unique ability to trunk multiple calls down one dataflow

  - Includes some extra signalling

  - Uses a single UDP port, so NAT friendly

  - Can use plaintext, MD5, or RSA key exchange for authentication

- IAX, although open source, is not a widely adopted standard

SIP

- SIP is *the* VoIP protocol these days - RFC 3261 (obsoletes RFC 2543)

- Original (simple!) draft created in 1996

- We'll be concentrating on SIP and largely ignoring the rest

    - It is worth playing around with IAX if you are going to be using Asterisk

- Largely ignored early on it's life (H.323 was used)

- Largely standard implementations of SIP now

- Not overly NAT friendly, although workarounds exist

- Worthy of a more in-depth look!

What is SIP?

"Session Initiation Protocol (SIP),

an application-layer control (signaling) protocol for creating, modifying, and terminating sessions with one or more participants"

(RFC 3261)

SIP Overview

- ASCII based signalling protocol

- Analogous to HTTP messages

- Works independent of the underlying network transmission protocol

- Provides mechanisms to:

    - Establish a session

    - Maintain a session

    - Modify and Terminate a session

SIP Overview

- Strength is it's simplicity and basic assumptions

- Component reuse

  - A child of SMTP and HTTP

  - SIP also uses MIME to carry extra information

  - Uses URI Eg: sip:jonnyphone@jonnynet.net

SIP Overview

- Scalable and robust protocol

    - Can offload various separate SIP functions to dedicated servers

    - Uses distributed architecture

- Very inter-operable protocol (well, these days it is!)

- Supports mobility through use of endpoint registration

- Uses RTP to carry media

SIP elements

- SIP User Agents

  - User Agent Clients (UAC) - the entity which initiates a call

  - User Agent Servers (UAS) - the entity which receives a call

- SIP Servers

  - Registrar server

  - Proxy server

  - Location server

  - Redirect server

SIP Registrar Server

- Users send registration requests to Registrar server

- Keeps track of client locations

- Supports various forms of authentication

- Often combined with the functionality of a Proxy server (Asterisk does this)

SIP Proxy Server

- Acts both as a server and a client

- Receives SIP messages, forwards to next SIP server

- Can perform functions such as Authentication, Authorisation, and Accounting (AAA)

- Provides network access control

- Requests are serviced internally or by passing them on to other servers.

- Interprets, rewrites or translates a request message before forwarding it.

SIP Messages

SIP Methods:

- INVITE – Initiates a call by inviting user to participate in session.
- ACK - Confirms that the client has received a final response to an INVITE request.
- BYE - Indicates termination of the call.
- CANCEL - Cancels a pending request.
- REGISTER – Registers the user agent.
- OPTIONS – Used to query the capabilities of a server.
- INFO – Used to carry out-of-bound information, such as DTMF digits.

SIP Responses:

- 1xx - Informational Messages
  - 180 ringing
- 2xx - Successful Responses
  - 200 OK
- 3xx - Redirection Responses
  - 302 Moved Temporarily
- 4xx - Request Failure Responses.
  - 404 Not Found
- 5xx - Server Failure Responses.
  - 503 Service Unavailable
- 6xx - Global Failures Responses.
  - 600 Busy Everwhere

SIP Messages

Informational

- 100 Trying
- 180 Ringing
- 181 Call forwarded
- 182 Queued
- 183 Session Progres

Success

- 200 OK

Redirection

- 300 Multiple Choices
- 301 Moved Perm.
- 302 Moved Temp.
- 380 Alternative Serv.

Request Failure

- 400 Bad Request
- 401 Unauthorised
- 403 Forbidden
- 404 Not Found
- 405 Bad Method
- 415 Unsupported Content
- 420 Bad Extensions
- 486 Busy Here

SIP Messages

Server Failure

- 504 Timeout
- 503 Unavailable
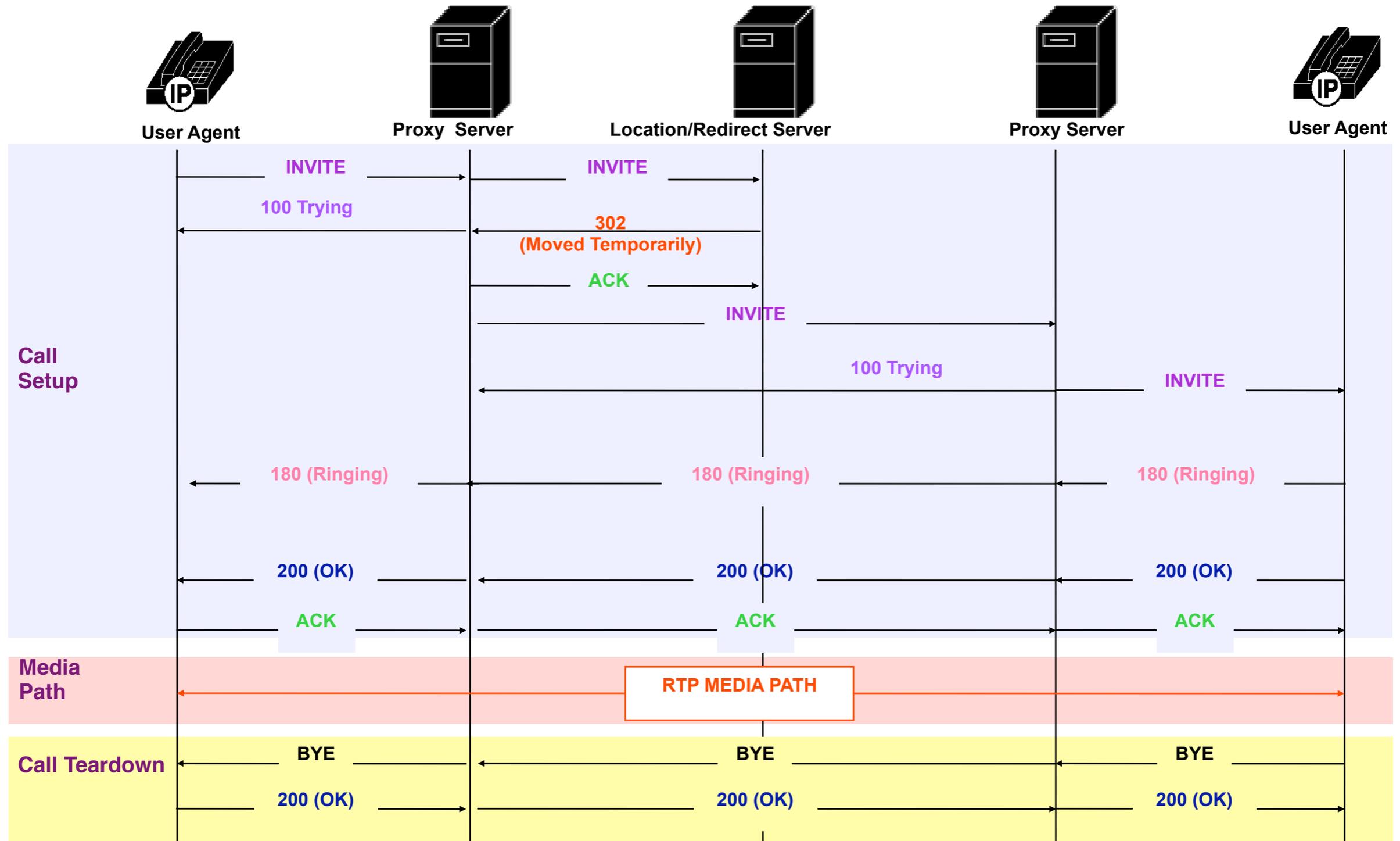- 501 Not Implemented
- 500 Server Error

Global Failure

- 600 Busy Everwhere
- 603 Decline
- 604 Doesn't Exist
- 606 Not Acceptable

SIP Addressing

- Can use SMTP style addressing

  - sip:jonnyphone@jonnynet.net

- Or E.164 (telephone number) addressing

  - sip:64212304323@jonnynet.net

Example SIP Call Flow

User Agent	Proxy Server	Location/Redirect Server	Proxy Server	User Agent

Call Setup

INVITE → INVITE →

100 Trying ←

← 302 (Moved Temporarily)

ACK →

INVITE →

100 Trying

← INVITE →

← 180 (Ringing) ← 180 (Ringing) ← 180 (Ringing)

← 200 (OK) ← 200 (OK) ← 200 (OK)

ACK → ACK → ACK →

Media Path

RTP MEDIA PATH

Call Teardown

← BYE ← BYE ← BYE

200 (OK) → 200 (OK) → 200 (OK) →

SIP Registration

```
Sip read:
REGISTER sip:203.114.148.130 SIP/2.0
Via: SIP/2.0/UDP 10.71.0.222:5060;rport;branch=z9hG4bK75D24E71C03111DB8A1300112476567E
From: Jonny test <sip:4989560@203.114.148.130>;tag=1675365723
To: Jonny test <sip:4989560@203.114.148.130>
Contact: "Jonny test" <sip:4989560@10.71.0.222:5060>
Call-ID: 7574F569C03111DB8A1300112476567E@203.114.148.130
CSeq: 19613 REGISTER
Expires: 1800
Authorization: Digest
username="4989560",realm="asterisk",nonce="25a752f4",response="ea87d99f48b43a97b39819e3fedbf8b8",uri="sip:
203.114.148.130"
Max-Forwards: 70
User-Agent: X-Lite release 1105x
Content-Length: 0

Transmitting (NAT) to 202.146.237.70:5060:
SIP/2.0 200 OK
Via: SIP/2.0/UDP
10.71.0.222:5060;branch=z9hG4bK75D24E71C03111DB8A1300112476567E;received=202.146.237.70;rport=5060
From: Jonny test <sip:4989560@203.114.148.130>;tag=1675365723
To: Jonny test <sip:4989560@203.114.148.130>;tag=as52d7bb4c
Call-ID: 7574F569C03111DB8A1300112476567E@203.114.148.130
CSeq: 19613 REGISTER
User-Agent: Asterisk PBX
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER
Expires: 1800
Contact: <sip:4989560@10.71.0.222:5060>;expires=1800
Date: Mon, 19 Feb 2007 14:54:56 GMT
Content-Length: 0
```

SIP Invite

```
Sip read:
INVITE sip:0212304323@203.114.148.130 SIP/2.0
Via: SIP/2.0/UDP 10.71.0.222:5060;rport;branch=z9hG4bKC19D7202C03111DB8A1300112476567E
From: Jonny test <sip:4989560@203.114.148.130>;tag=1386353914
To: <sip:0212304323@203.114.148.130>
Contact: <sip:4989560@10.71.0.222:5060>
Call-ID: C06D0E06-C031-11DB-8A13-00112476567E@10.71.0.222
CSeq: 32821 INVITE
Max-Forwards: 70
Content-Type: application/sdp
User-Agent: X-Lite release 1105x
Content-Length: 205

v=0
o=4989560 81389423 81389572 IN IP4 10.71.0.222
s=X-Lite
c=IN IP4 10.71.0.222
t=0 0
m=audio 8000 RTP/AVP 3 101
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=sendrecv
```

SIP Invite - Response

```
Reliably Transmitting (NAT):
SIP/2.0 200 OK
Via: SIP/2.0/UDP
10.71.0.222:5060;branch=z9hG4bKC242AB12C03111DB8A1300112476567E;received=202.146.237.70;rport=5060
From: Jonny test <sip:4989560@203.114.148.130>;tag=1386353914
To: <sip:0212304323@203.114.148.130>;tag=as77d3c840
Call-ID: C06D0E06-C031-11DB-8A13-00112476567E@10.71.0.222
CSeq: 32822 INVITE
User-Agent: Asterisk PBX
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER
Contact: <sip:0212304323@203.114.148.130>
Content-Type: application/sdp
Content-Length: 269

v=0
o=root 26612 26612 IN IP4 203.114.148.130
s=session
c=IN IP4 203.114.148.130
t=0 0
m=audio 19918 RTP/AVP 8 0 3 101
a=rtpmap:8 PCMA/8000
a=rtpmap:0 PCMU/8000
a=rtpmap:3 GSM/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-16
a=silenceSupp:off - - - -
```

SIP in Detail

- There's much more to SIP than we can possibly hope to cover here

  - Go read the RFC!