

TESI DI LAUREA in INGEGNERIA INFORMATICA e AUTOMATICA

*Realizzazione di un VoIP gateway
su piattaforma embedded Linux*



Relatore:

Prof. Daniele Frigioni

Correlatore:

Ing. Fabio Antonini

Candidato:

Davide Di Gesualdo

Obiettivo della tesi

Realizzare un **VoIP** gateway utilizzando un **sistema embedded Linux** sulla **piattaforma** Intel IXDPG425 calando, all'interno dell'architettura software Intel IXP400 Sw, un modulo Kernel-**RTP**, in modo da ottimizzarne le prestazioni

Sommario

- **VoIP:** Voice over IP
 - **RTP:** Realtime Transfer Protocol
 - Sistemi embedded Linux
 - Piattaforma di sviluppo
 - Adattamento krtpd alla piattaforma
 - Conclusioni
-
-

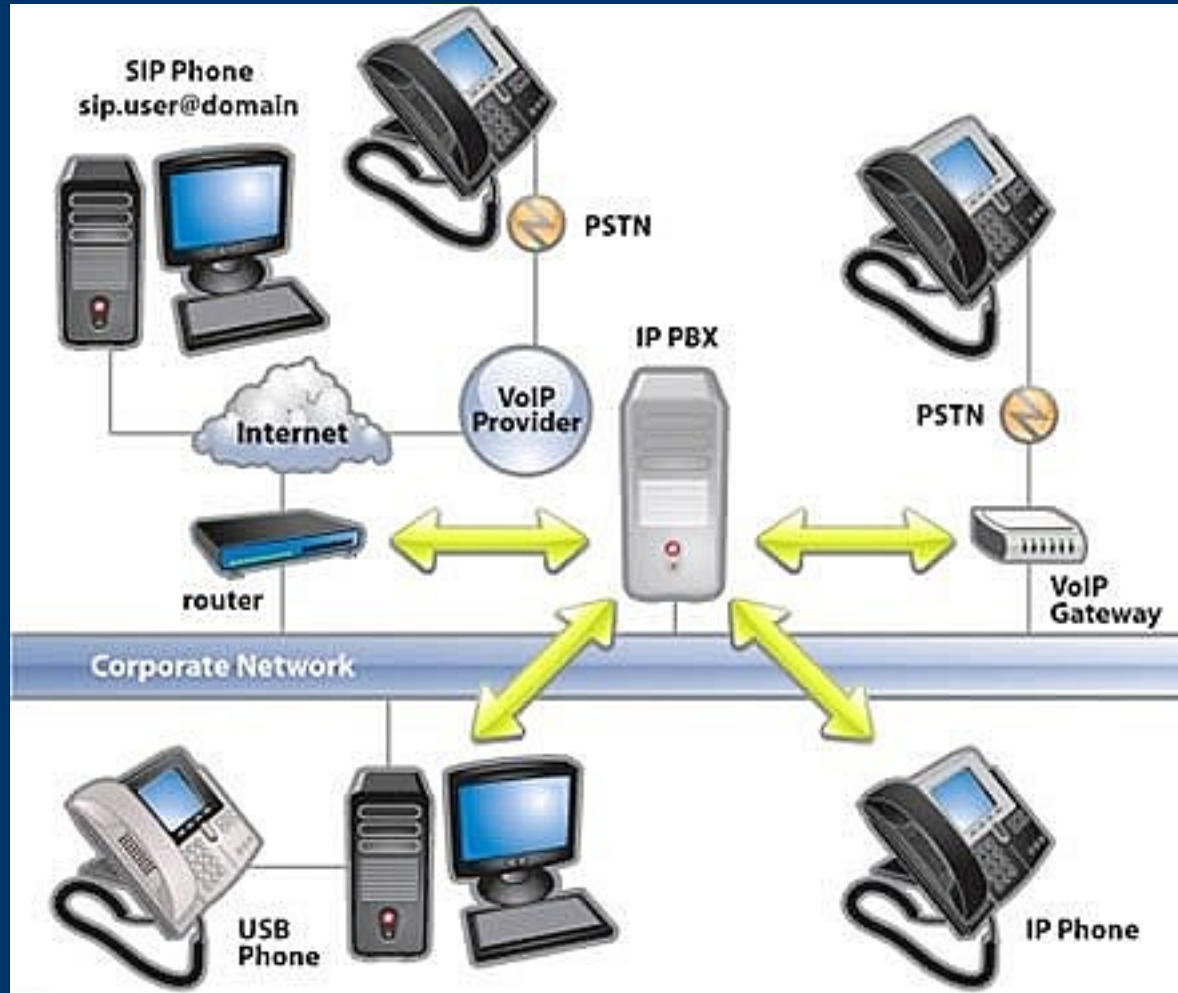
VoIP: cos'è

Il VoIP è una tecnologia che rende possibile effettuare una conversazione telefonica sfruttando una connessione Internet o un'altra rete dedicata che utilizza il protocollo IP, anziché passare attraverso la rete telefonica tradizionale (PSTN)

VoIP: vantaggi

- I dati hanno superato la voce come traffico principale in molte reti realizzate per la voce
 - Le reti PSTN non sono in grado di creare e fornire nuove funzionalità con la dovuta rapidità
 - Le informazioni dati/voce/video non possono convergere sulle attuali reti PSTN
 - Vantaggio economico
-
-

VoIP: esempio



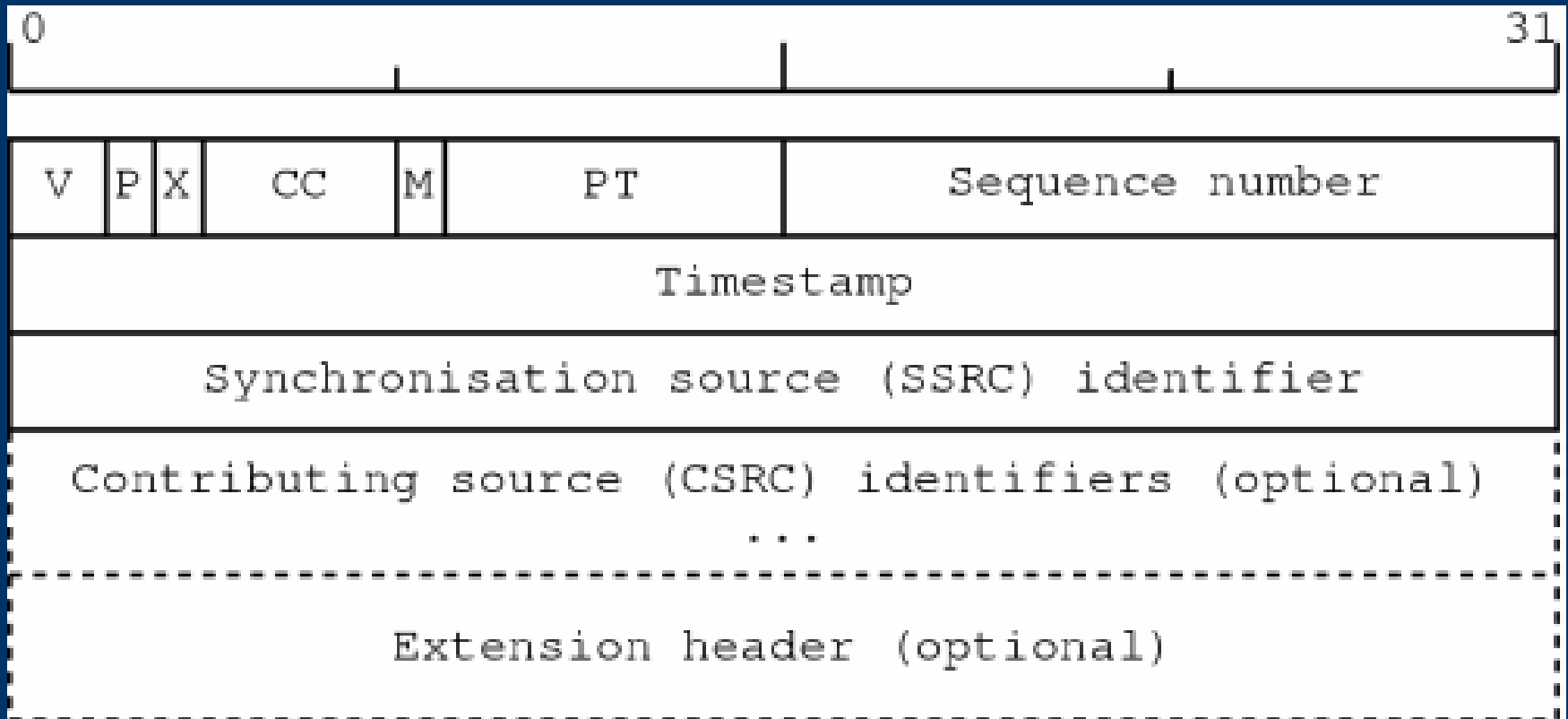
VoIP: problemi

- Ritardo o latenza
 - Tremolio o jitter
 - Eco
 - Perdita di pacchetti
 - Compressione della voce
-
-

RTP: cenni

- Standard per la trasmissione del traffico sensibile ai ritardi lungo reti a pacchetti
 - Basato sul protocollo UDP
 - Affiancato al protocollo RTCP
 - Customizzabile a seconda dell'applicazione
-
-

RTP: intestazione



Sistemi embedded Linux

Generalità

- Sistemi elettronici a microprocessore progettati appositamente per una determinata applicazione, spesso con una piattaforma hardware ad hoc
- Utilizzano kernel o distribuzioni Linux



Sistemi embedded Linux

Realizzazione del target

- Determinare i componenti del sistema
 - Configurare e compilare il kernel
 - Creare il filesystem radice
 - Scelta bootloader e configurazione
-
-

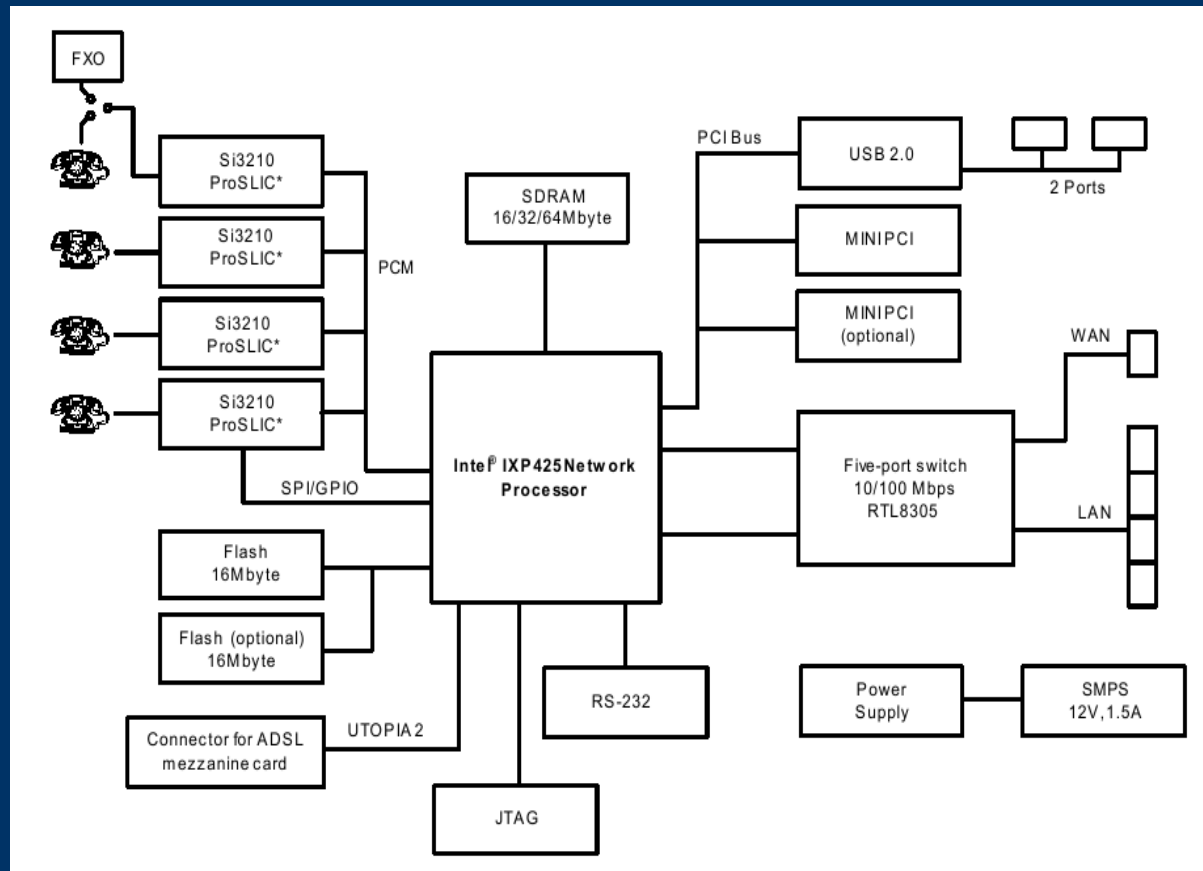
Sistemi embedded Linux

Strumenti di sviluppo

- Configurare un workspace
 - Toolchain cross-platform
 - ~~Librerie~~ Librerie C (glibc, uClibc, diet libc)
 - Altri linguaggi di programmazione (Java, Perl, Python)
 - IDE (Eclipse, NetBeans, KDevelop, Source Navigator)
-
-

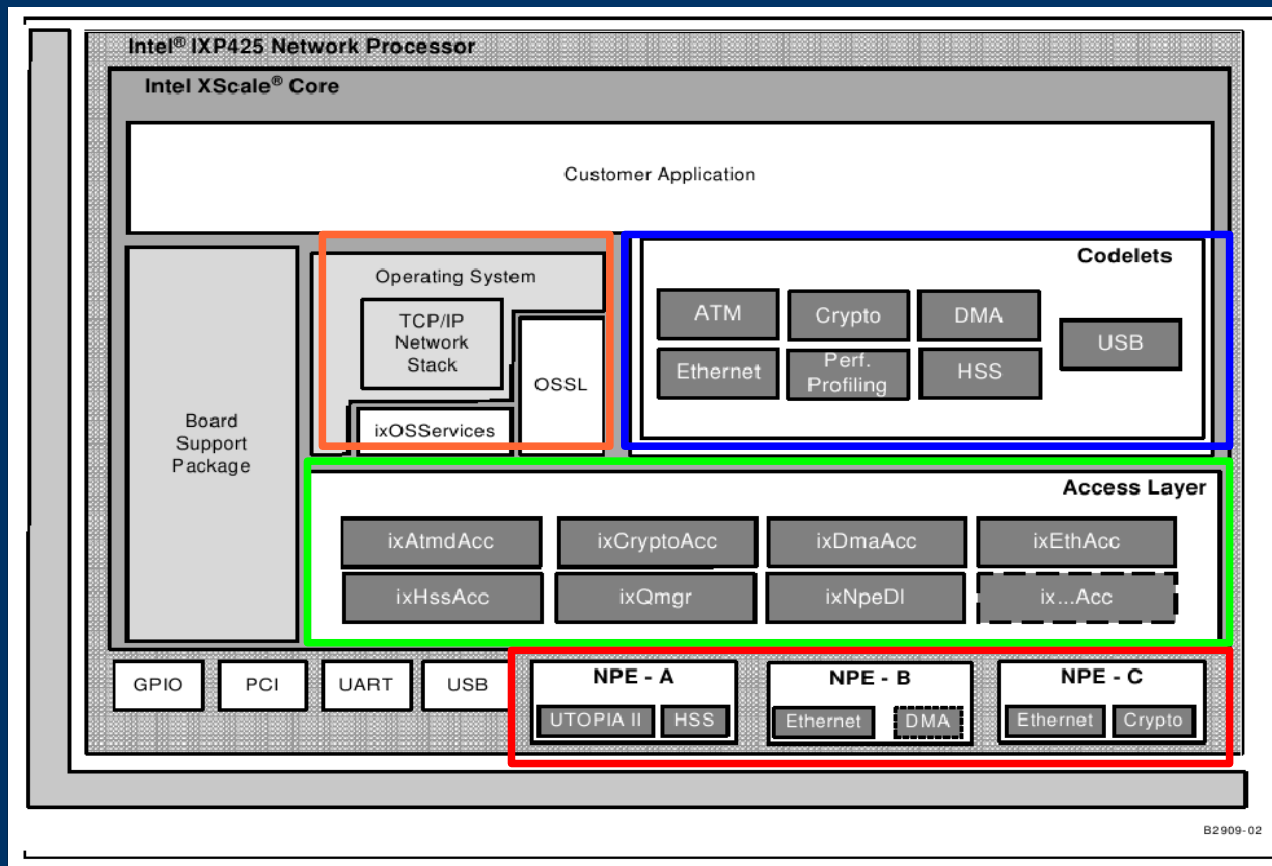
Piattaforma di sviluppo

Hardware: Intel[®] IXDPG425



Piattaforma di sviluppo

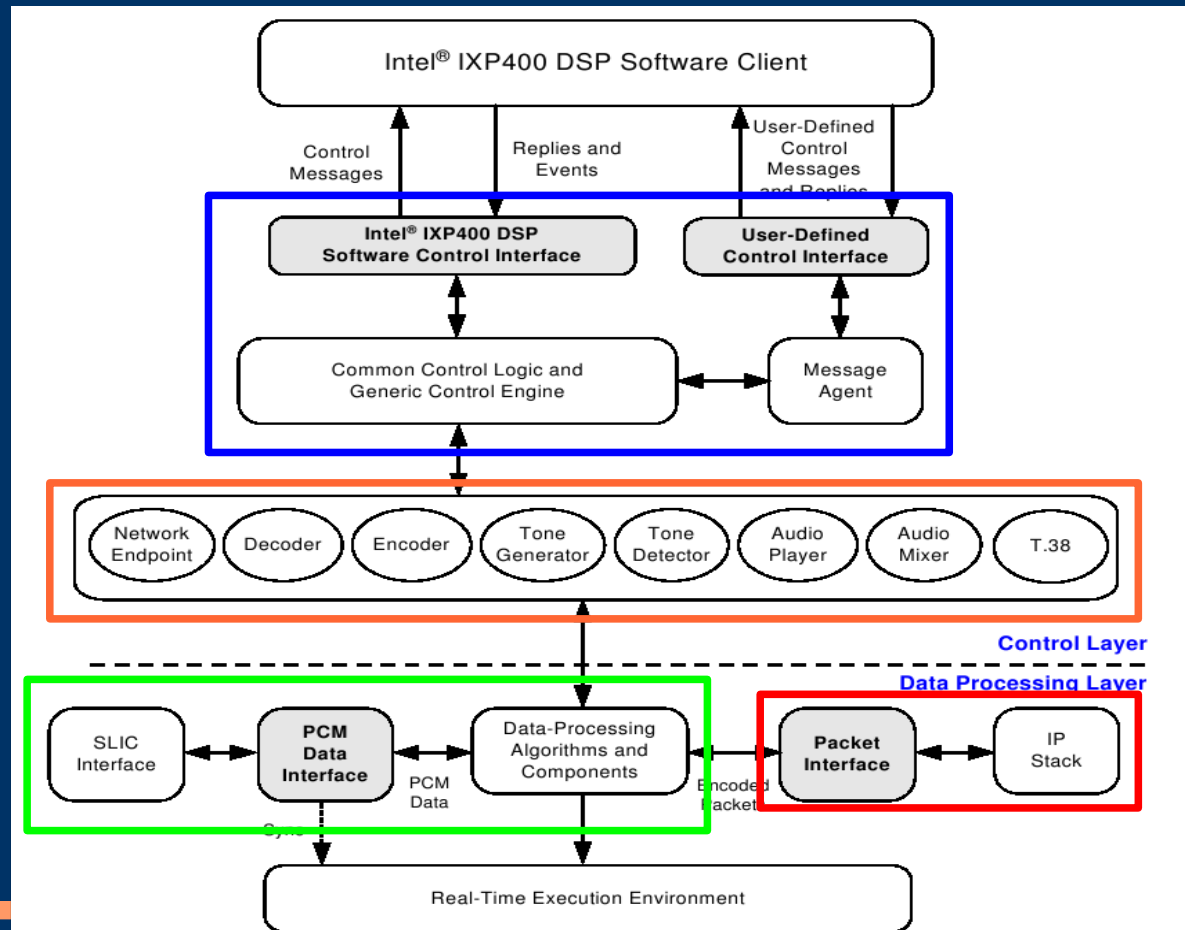
Software: Intel[®] IXP400 Software v1.4



Piattaforma di sviluppo

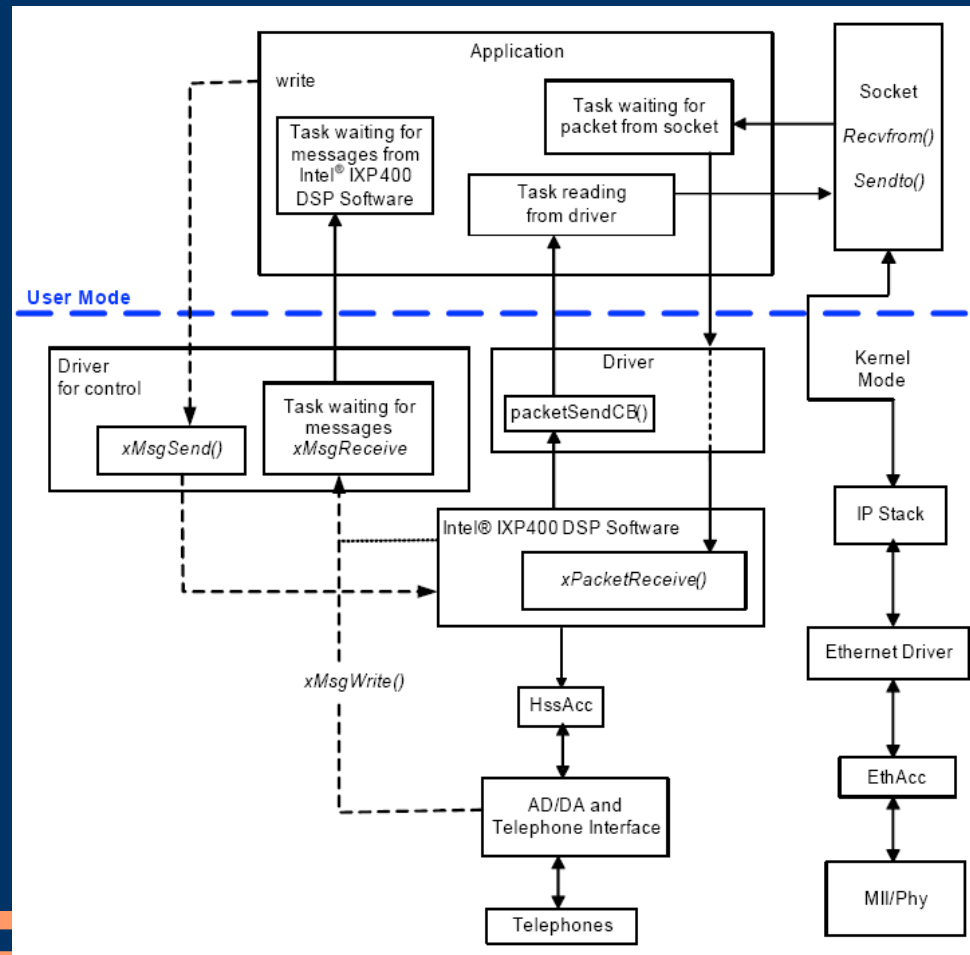
Software: Intel[®] IXP400 DSP Software

v2.6.2



Piattaforma di sviluppo

Flusso dati rete/DSP nella piattaforma software



Scenario

USER APPLICATION

user space

kernel space



DSP

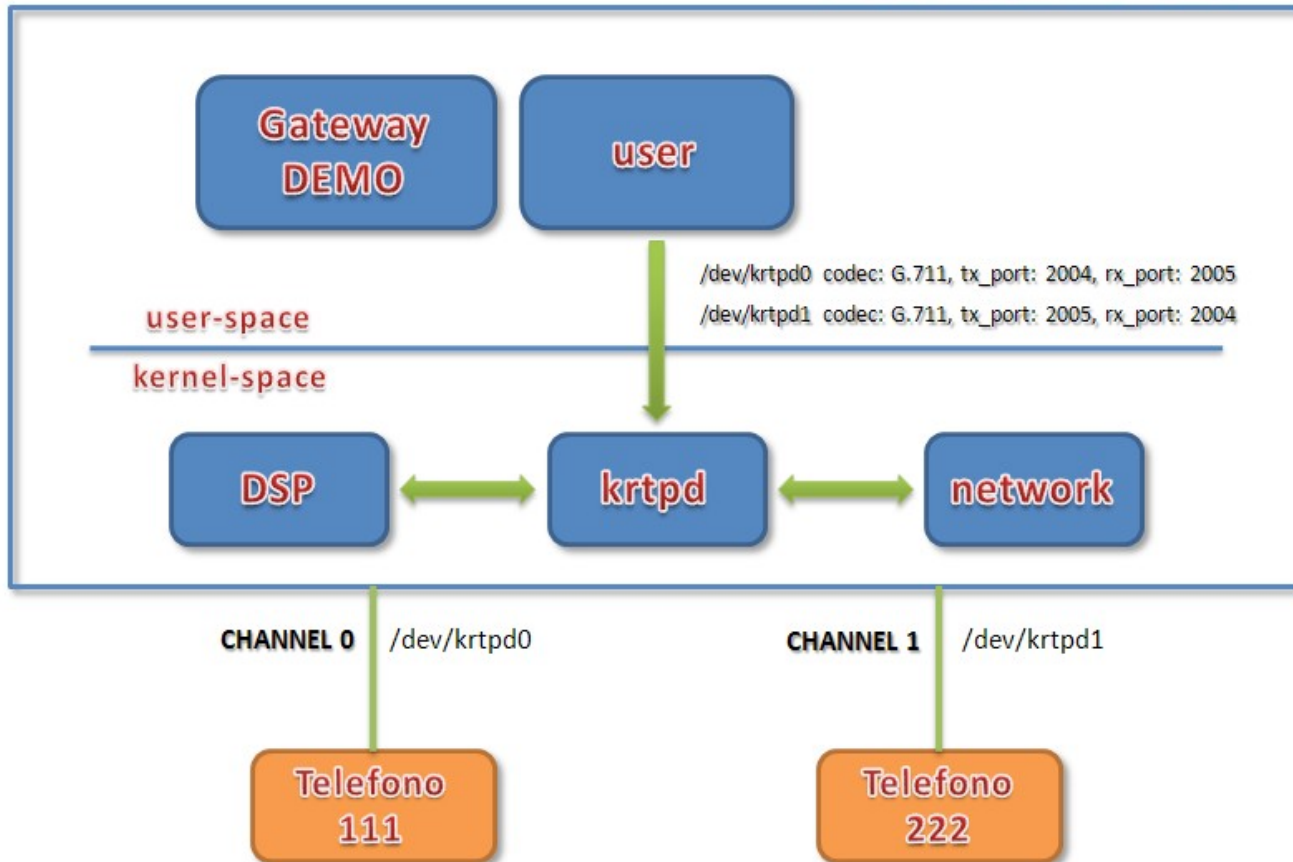
krtpd

NETWORK



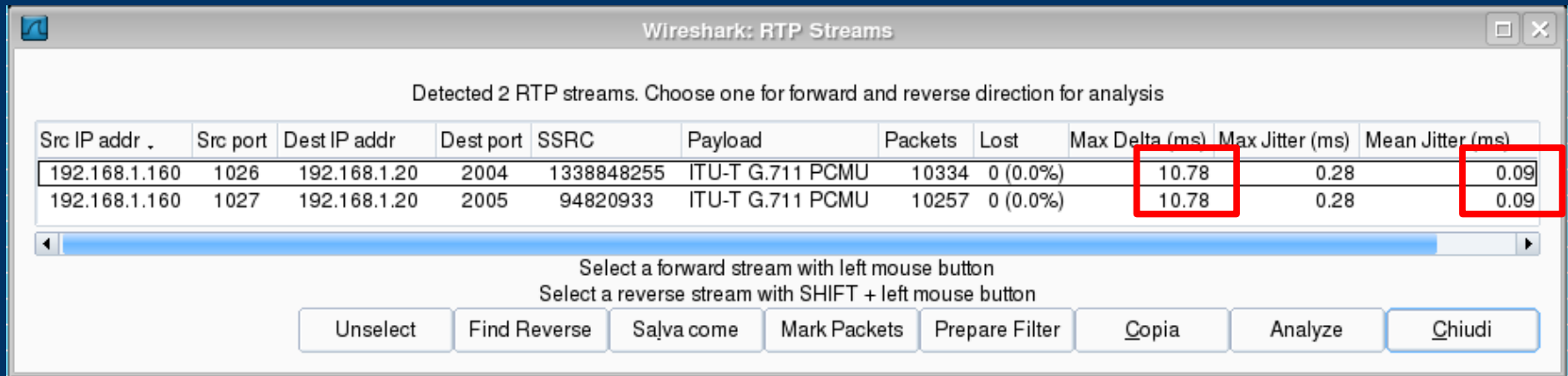
Dimostrazione

Intel® IXDPG425



Risultati ottenuti

Sintesi delle informazioni sugli stream RTP catturati



Wireshark: RTP Streams

Detected 2 RTP streams. Choose one for forward and reverse direction for analysis

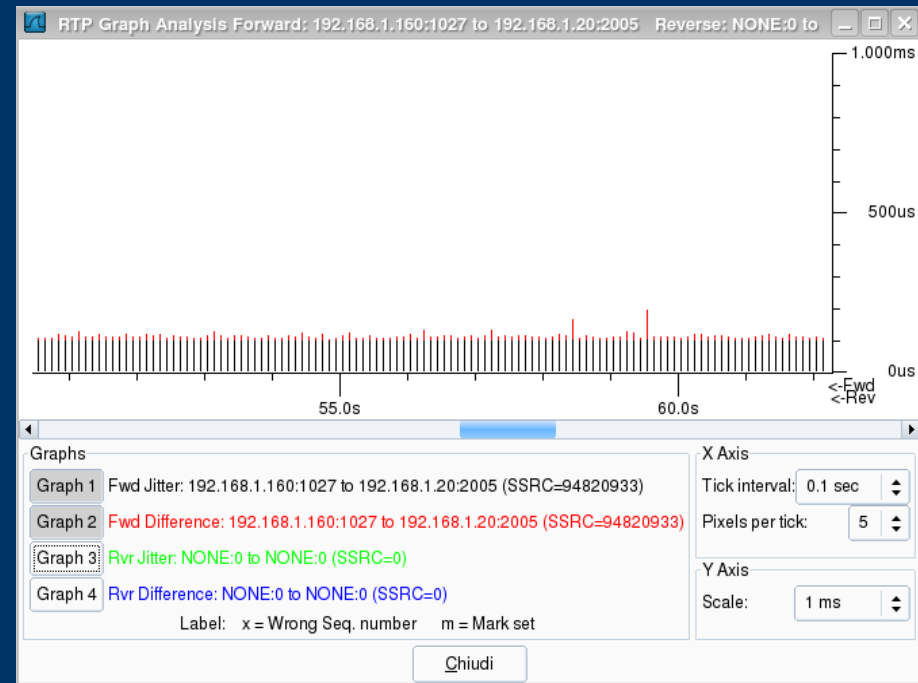
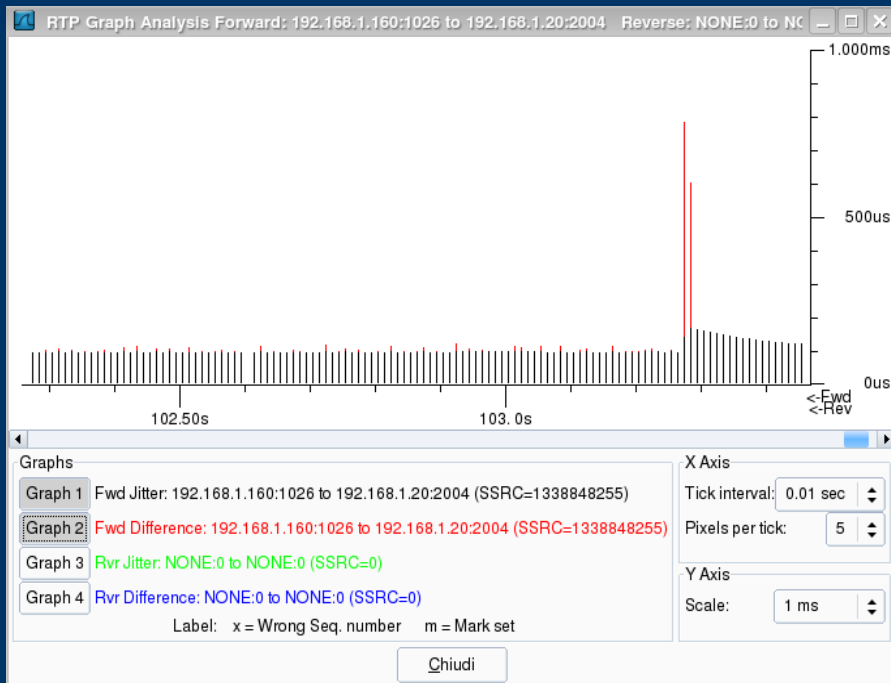
Src IP addr	Src port	Dest IP addr	Dest port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter (ms)	Mean Jitter (ms)
192.168.1.160	1026	192.168.1.20	2004	1338848255	ITU-T G.711 PCMU	10334	0 (0.0%)	10.78	0.28	0.09
192.168.1.160	1027	192.168.1.20	2005	94820933	ITU-T G.711 PCMU	10257	0 (0.0%)	10.78	0.28	0.09

Select a forward stream with left mouse button
Select a reverse stream with SHIFT + left mouse button

Unselect Find Reverse Salva come Mark Packets Prepare Filter Copia Analyze Chiudi

Risultati ottenuti

Andamento temporale dei pacchetti scambiati



Conclusioni

- Jitter medio rilevato: 0.09ms
 - Eliminato l'accesso esclusivo alle interfacce di rete da parte del DSP software
 - Utilizzo del krtpd in applicativi reali: linphone, PBX Asterisk
-
-