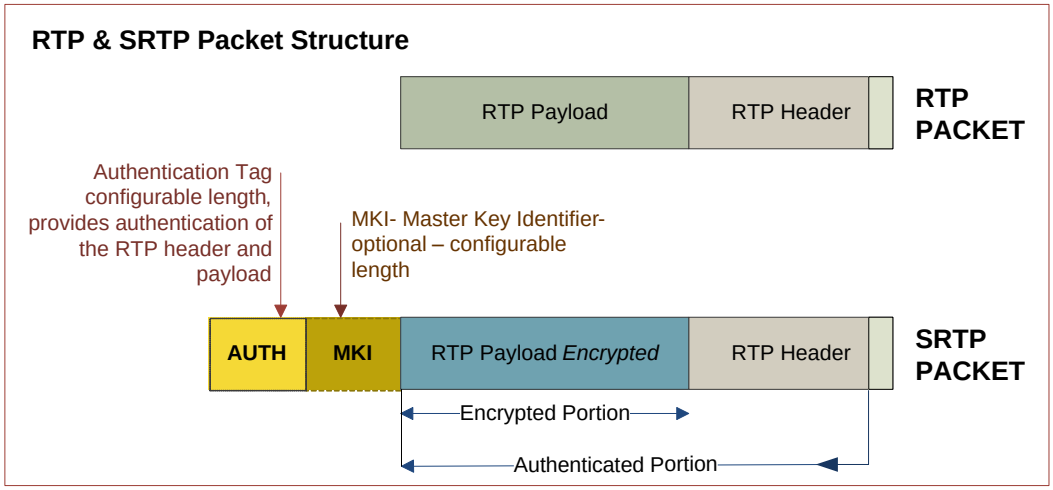
RTP Series

The Real-Time Transport Protocol (RTP) is a networking protocol that is used to transport real-time media data streams such as voice and video over packet networks. This protocol is an industry standard that is defined by IETF RFC 3550.



FEATURES

- Multi-channel capable.
- Functions are C-callable.
- Support of RTP version 2 protocol as defined in RFC 3550
- Independence from underlying protocol stack.
- Minimum count of consecutively increasing sequence numbers prior to playout.
- Built in jitter buffer
- Automatic timestamp synchronization.
- Multiple independent RTP streams
- Re-entrant routines callable by multiple processing threads
- TI eXpressDSP™ Algorithm Interoperability Standard (xDAIS) Compliant
- Secure variants include support for:
  - Authentication Algorithm Types: HMAC-SHA1 and MD5
  - Key Definition Schemes: PSK, MKI, and FT
  - Encryption types: CM, F8

PRODUCT DESCRIPTION

Adaptive Digital Technologies' Real Time Protocol (RTP) software provides transport layer functionality for real-time applications communicating over an IP network. This product also contains a built in, configurable, jitter buffer to compensate for network delays, out-of-order packets, and lost packets.

Adaptive Digital's implementation of the RTP protocol is designed to provide fully re-entrant modules to allow multiple RTP streams to be processed concurrently. User supplied callback and support routines are used to allow the RTP software to be easily adapted to the application environment. Memory allocation for the packets stored within the jitter buffer is dynamic and is accomplished via a call to a user definable memory allocation support routine. To allow flexibility in interfacing with differing network stack mechanisms, the mechanism to send data over the network is accomplished via a user definable callback routine.

The built-in jitter buffer provides a storage mechanism for inbound packets. The RTP module stores incoming packets into the jitter buffer at the time of reception. The jitter buffer uses the RTP header timestamp and sequence number to position out-of-order packets correctly within the jitter buffer. Packets remain stored in the jitter buffer until they are ready for delivery to the receiving application.

RTP, RTCP, AND THEIR SECURE COUNTERPARTS

The successful transport of real-time voice and video data necessitates the use of transport protocols that are different from the traditional protocols, such as TCP and UDP, that are typically used for the transport of non-real-time data. There are three primary differences:

Voice and Video need to be "played out" in a continuous fashion in spite of the bursty nature of packet networks.

When a packet is lost, late, or received in error, there is no time to request a retransmission.

Packets that are received out of sequence must be re-sequenced so that they can be played out in the correct order.

RTP is specifically designed to handle the play-out requirements of real-time media streams through the use of time stamps and jitter buffering. Due to the real-time nature of the data streams, where requesting retransmissions is too costly in time, RTP is typically used in conjunction with UDP to provide low-overhead network communications between two end-points.
RFC 3550 identifies two components to the real-time transport: data transport and control. Data transport is handled by RTP while Real Time Control Protocol (RTCP) handles control. RTCP, which can be used to help scale the network traffic to the available bandwidth, is optional.

An RTP packet identifies the media payload type (format) and its source. It also includes time stamps and sequence numbers that are used by the play-out side to handle lost or out of sequence packets. RTP provides for the use of multiple streams as in the case of a system that transmits both voice and video. The payload in an RTP payload contains the encoded voice or video information. The use of dynamically defined payload types allows RTP packets to carry virtually any type of media format.

RTCP is used to keep track of packet reception statistics and to provide supplementary information (user and domain name, e-mail address, phone number, etc.) about the source of the media data stream. It also assists with the synchronization of multiple RTP streams. While RTP packets are sent at high rates to handle the real-time media streams, RTCP packets can normally be sent much less often – a typical rate being every few seconds.

A secure transmission feature, as defined in IETF RFC 3711 is also available for these protocols. When security is used, an “S” prefaces the acronyms: SRTP, and SRTCP. When security is being used, the packet payloads are encrypted.

AVAILABILITY

ADT **RTP** is available in transportable “C” source code format as well as in library object format on all the Texas Instruments TMS320™ DSPs, and TNETV™ family of VoIP processors.

Platform	Memory Model	Endian	Code Gen Tool Version
TI TMS320C64x+/C66x/C674x	L3	Little	N/R
TI TMS320C64x	L3	Little	N/R
TI TMS320C55x	L3	Little	N/R
Win32 (lib & dll)	N/A	Little	VS2010
Arm9e	N/A	Little	Code Sourcery Linux 2011_09-70
ARMv7A | ARMv8A	N/A	Little	Code Sourcery Linux 2011_09-70
ARMv7M	N/A	Little	N/R
i686	N/A	Little	gcc

SPECIFICATIONS

TI TMS320C6000

Peak CPU utilization occurs when a new master encryption key is required. Normal voice over IP applications require only a single key for the duration of a conversation; in this case, the peak occurs only at the start of an RTP stream.

C64x & C64x+ MIPS

CPU UTILIZATION

RTP		Average Frame Rate			Peak Frame Rate		
Processor	Direction	10 msec	20 msec	30 msec	10 msec	20 msec	30 msec
C64x	Rx	0.26	0.14	0.10	0.43	0.22	0.15
C64x	Tx	0.2	0.1	0.07	0.2	0.1	0.07
C64x+	Rx	0.18	0.09	0.05	0.54	0.27	0.26
C64x+	Tx	0.11	0.06	0.04	0.21	0.11	0.07

MEMORY REQUIREMENTS

Program memory is shared between all streams. All function calls are re-entrant and all stream share the same program memory.

C64x & C64x+

	Memory Bytes	
Processor	Program Memory	Instance* Data Memory
C64x	9652	172
C64x+	8324	172

* + Jitter Buffer Size

C55x

RTP		Average Frame Rate			Peak Frame Rate		
Processor	Direction	10 msec	20 msec	30 msec	10 msec	20 msec	30 msec
C55x	Rx	0.52	0.28	0.20	0.86	0.44	0.30

	Memory Bytes	
Processor	Program Memory	Data Memory
C55x	7056	280

ARM

ARMv7A CORTEX-A8/A9/A12/A15/A19 | ARMv8A Cortex-A53/A57

CPU UTILIZATION & MEMORY REQUIREMENTS

All Memory usage is given in units of bytes.

	Average MIPS*	Program Memory	Instance* Data Memory
Tx	.20	8.6k	172
Rx	.20		

* + Jitter Buffer Size

ARMv7M CORTEX-M3/M4

MEMORY REQUIREMENTS

All Memory usage is given in units of bytes.

	Program Memory	Data Memory	Instance* Data Memory
RTP	6.4 K	108	200
RTP + RTCP	15.7 K	124	660

* + Jitter Buffer Size

CPU UTILIZATION

RTP	Average MIPS*		
Function	10 msec	20 msec	30 msec
Tx	0.13	0.08	0.06
Rx	0.22	0.12	0.09

ARM9e/ARM11

CPU UTILIZATION & MEMORY REQUIREMENTS

All Memory usage is given in units of bytes.

	Average MIPS**	Program Memory	Instance* Data Memory
Tx	.20	8.6k	172
Rx	.10		

**Actual MIPS will vary dependent upon: packet size, frame rate, and application supplied callback routines.
* + Jitter Buffer Size

PC/Windows

Win Static Lib

CPU UTILIZATION & MEMORY REQUIREMENTS

All Memory usage is given in units of bytes.

	Average MIPS	Program Memory	Instance* Data Memory
Tx	<0.33**	8.6k	172
Rx	<0.33*		

**Value too small to measure accurately, less than 0.33
* + Jitter Buffer Size

Win DLL

CPU UTILIZATION & MEMORY REQUIREMENTS

All Memory usage is given in units of bytes.

	Average MIPS	Program Memory	Instance* Data Memory
Tx	<0.33**	38k	172
Rx	<0.33		

**Value too small to measure accurately, less than 0.33
* + Jitter Buffer Size

LINUX

Linux i686

CPU UTILIZATION & MEMORY REQUIREMENTS

All Memory usage is given in units of bytes.

	Average MIPS	Program Memory	Instance* Data Memory
Tx	<0.33**	8.6k	172
Rx	<0.33		

**Value too small to measure accurately, less than 0.33
* + Jitter Buffer Size

We specify MIPS (Millions of Instructions Per Second) as MCPS (Millions of Instruction Cycles Per Second). Unless otherwise specified, peak MIPS are indicated.

TERMINOLOGY

Acronyms

RTP - Real-time Transport Protocol

SRTP - Secure Real-time Transport Protocol

AES CM - Advanced Encryption Standard counter mode

AES-f8 - AES in f8-mode, Universal Mobile Telecommunications System (UMTS) 3G mobile networks use AES-f8.

MKI - Master key identifier

HMAC - Hashed message authentication

MD5 - Message Digest 5 *is a widely used cryptographic hash function with a 128-bit hash value*

Deliverables

The deliverable items are platform dependent. In general, there is one library. (Sometimes multiple variants of the library are included in the deliverables.) There are also header files, some of which are specific to the product and others are common across many of Adaptive Digital's products. Also included in the deliverables is product documentation, which includes a users guide and usually includes release notes and a data sheet. Sample/test code may be included as well.

Adaptive Digital is a member of the Texas Instruments Developer Network, and ARM Connected Community.

CONTACT INFORMATION

Web: www.adaptivedigital.com
Email: information@adaptivedigital.com
Tel: 610.825.0182 ~ Toll Free: 1.800.340.2066
Fax: 610.825.7616
Address: 525 Plymouth Road, Suite 316
Plymouth Meeting, PA 19462