Embedded systems' transport protocol choosing for modelling over the SpaceWire model

Ilya Korobkov

ilya.korobkov@guap.ru SUAI

Task

▸ To implement simultaneous operation of multiple applications over SpaceWire Network with streaming data transfer (sensors, ADCs, video streams and etc)

Common features:

- ▸ continuous generation of data streams
- ▸ data chunks has the fixed length
- ▸ periodicity of data generation
- ▸ corrupted and lost data aren't retransmitted (in most cases it's impossible)
- ▸ receiver can't stop generation of data by the source instantaneously
- ▸ support of multiple coherent data streams

Review of existing protocols 1/5

Remote **M**emory **A**ccess **P**rotocol is to support reading from and writing to memory in a remote SpaceWire node
Using: system administration, gathering of information, setting/checking device parameters, casual data transmissions

Packet **T**ransfer **P**rotocol is designed to packet transfer over SpaceWire Network
- ▸ packets have a variable or fixed length
- ▸ unidirectional data transfer without acknowledgments
- ▸ in case data were lost – doesn't repeat them
- ▸ no periodicity of data sending
- ▸ no data verification

Review of existing protocols 2/5

Real-**T**ime **T**ransport **P**rotocol is aimed for end-to-end, real-time, transfer of stream data (interactive audio and video) over IP networks
- ▶ end-to-end delivery of real-time data
- ▶ doesn't provide any mechanism to ensure timely delivery
- ▶ no data flow control
- ▶ support multiple data transfer at the cost of multicast
- ▶ no handshake for connection setup and teardown

Real-**T**ime **S**treaming **P**rotocol is a client-server application protocol to enable controlled delivery of streamed multimedia data over IP networks and to control audio-video streams
- ▶ requests have text format, like HTTP PLAY rtsp://server/test.mpg RTSP/1.0
- ▶ server and client can issue requests
- ▶ server has to maintain methods SETUP, TEARDOWN.
- ▶ doesn't provide data transfer (it's a server's functionality)

Review of existing protocols 3/5

Stream **C**ontrol **T**ransmission **P**rotocol is a reliable universal transport protocol for use on IP networks
- ▶ stable delivery of data (acknowledgments, retransmission, flow control)
- ▶ multi-homing (some network interfaces)
- ▶ multi-streaming
- ▶ reliable connection (4-way handshake)
- ▶ configurable unordered delivery

Structured **S**tream **T**ransport **P**rotocol is a new transport protocol designed to combine asynchronous multiple data transfer (downloading web pages and playing audio-video streams at once)
- ▶ no 3-way handshake on startup or TIME-WAIT on close
- ▶ multiplexes many application streams onto one network connection
- ▶ supports messages/datagrames of any size
- ▶ dynamic prioritization of streams
- ▶ optional cryptographic security (comparable to SSL)

Resource **R**eservation **P**rotocol is a transport protocol, provides a special end-to-end quality of service (QoS) for data flows: RSVP reserves necessary resources along the transmission paths so that the requested bandwidth can be available when the transmission actually takes place

- ▸ multi-streaming
- ▸ unidirectional reservation of resources
- ▸ support special QoS to multicast and unicast
- ▸ data flow control

Datagram **C**ongestion **P**rotocol – a transport protocol, provides bidirectional unicast connections to transfer large amounts of data (streamed media)

- ▸ flow of datagrams, with acknowledgments, no retransmission
- ▸ congestion control
- ▸ reliable handshake for connection setup and teardown
- ▸ reliable negotiation of connection parameters

Review of existing protocols 5/5

Streaming **T**ransport **P**rotocol – a new transport protocol to process streaming data over SpaceWire networks. STP is in progress.

- ▸ reliable handshake for connection setup and teardown (3-way handshake)
- ▸ multiple coherent data streams
- ▸ fixed length of transmitted data
- ▸ periodical continuous data transfer
- ▸ data delivery without acknowledgements, no retransmission
- ▸ flow control
- ▸ STP is efficient for continuous transmission of large amounts of data

Comparison of protocols

Protocol	Periodicity of data generation	Multi-streaming coherent data	Flow control	Handshake for connection setup	Small overheads with data delivery
RMAP	–	–	–	–	–
PTP	–	–	–	–	+
RTP	–	+	–	–	–
RTSP	+	+	+	+	–
SCTP	–	+	+	+	–
SSTP	–	+	+	–	–
RSVP	–	+	+	+	–
DCCP	–	+	+	–	–
STP	+	+	+	+	+

Results

- Review of existing streaming transport protocols

- STP is chosen, because it covers most of all requirements of SpaceWire applications

- STP model (according to the specification)
  The model is used for testing ability
  of STP work with SpaceWire
  protocol stack

- 2 critical cases were found
   (not described in specification)

- The found bugs were shared
   with STP workgroup

THANK Y OU