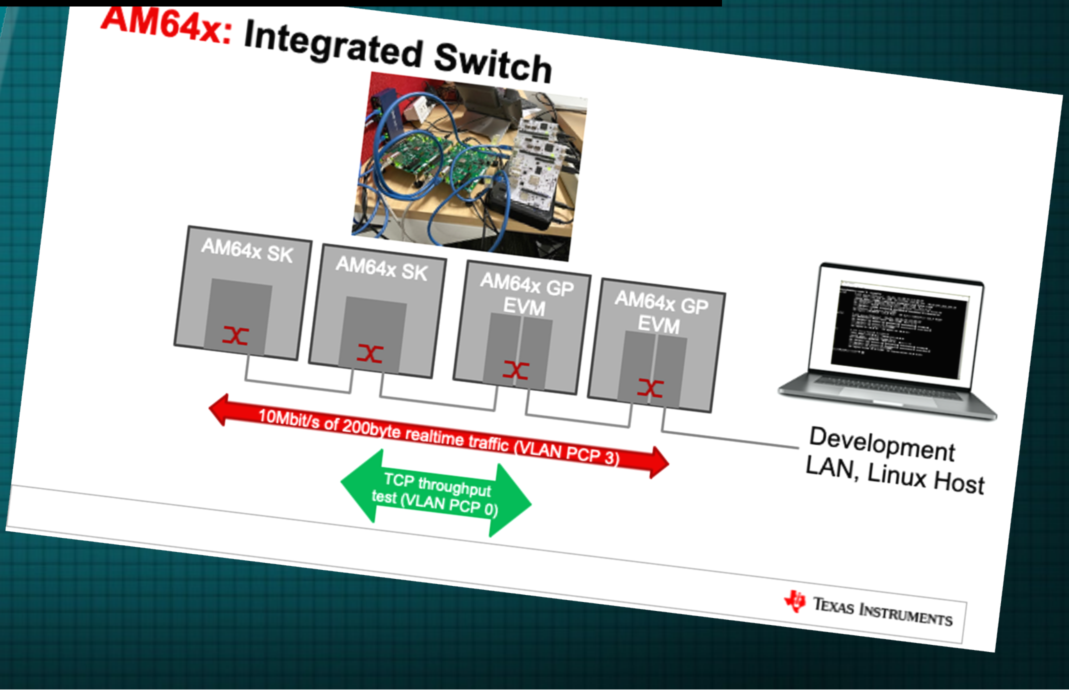
Linux Time Sensitive Networking

REAL TIME NETWORKING CUT-THROUGH, TIME SYNC, AND PREEMPTION



AM64x: Integrated Switch

AM64x SK

AM64x SK

AM64x GP EVM

AM64x GP EVM

Development LAN, Linux Host

10Mbit/s of 200byte realtime traffic (VLAN PCP 3)

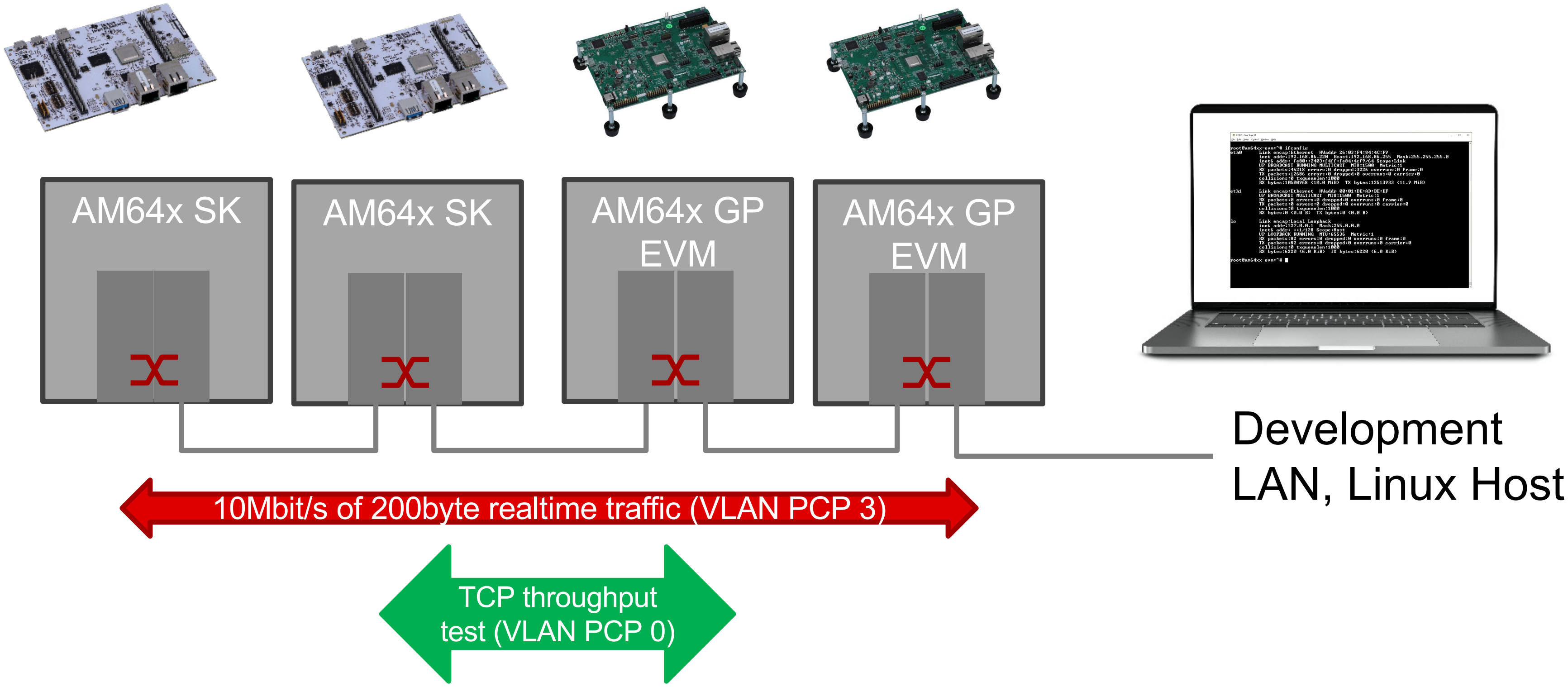
TCP throughput test (VLAN PCP 0)

Texas Instruments

Agenda

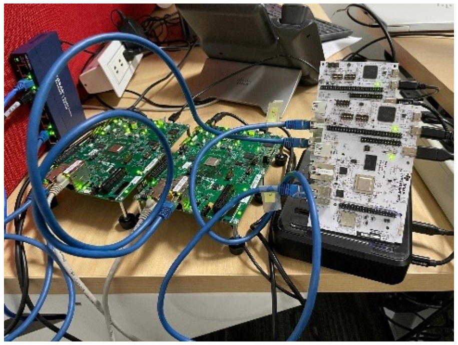
- Demo overview
- AM64x TSN features
- Demo
  - Configure cut-through switching with preemption
  - IEEE802.1AS-2011 using ptp4l
  - Express and background traffic
- Q & A

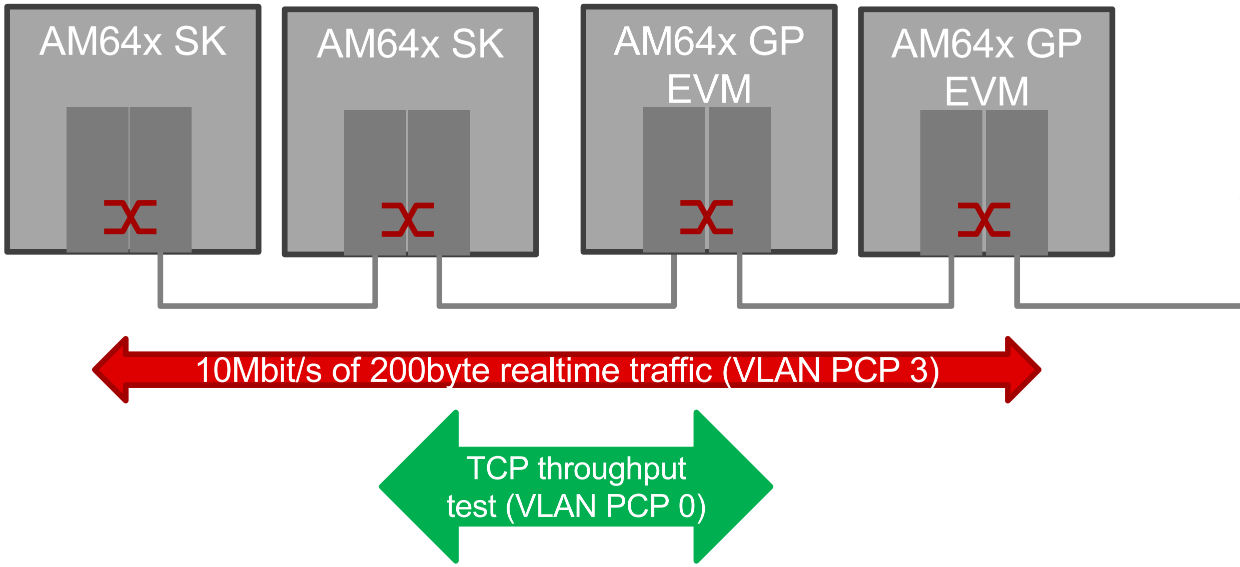
AM64x: Integrated Switch

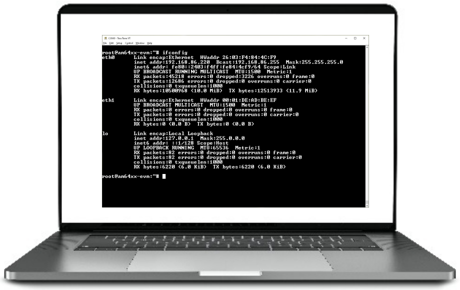

AM64x SK
AM64x SK
AM64x GP EVM
AM64x GP EVM
Development LAN, Linux Host
10Mbit/s of 200byte realtime traffic (VLAN PCP 3)
TCP throughput test (VLAN PCP 0)
File Edit Setup Control Window Help

```
root@am64xx-evm:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 26:03:F4:84:4C:F9
          inet addr:192.168.86.220  Bcast:192.168.86.255  Mask:255.255.255.0
          inet6 addr: fe80::2403:f4ff:fe84:4cf9/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:45218 errors:0 dropped:3226 overruns:0 frame:0
          TX packets:12686 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:10500968 (10.0 MiB)  TX bytes:12513933 (11.9 MiB)

eth1      Link encap:Ethernet  HWaddr 00:01:DE:AD:BE:EF
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:82 errors:0 dropped:0 overruns:0 frame:0
          TX packets:82 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6220 (6.0 KiB)  TX bytes:6220 (6.0 KiB)

root@am64xx-evm:~# 
```

AM64x: Integrated Switch




AM64x SK
AM64x SK
AM64x GP EVM
AM64x GP EVM
10Mbit/s of 200byte realtime traffic (VLAN PCP 3)
TCP throughput test (VLAN PCP 0)



```
root@am64xx-evm:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 26:03:F4:84:4C:F9
          inet addr:192.168.86.220  Bcast:192.168.86.255  Mask:255.255.255.0
          inet6 addr: fe80::2403:f4ff:fe84:4cf9/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:45218 errors:0 dropped:3226 overruns:0 frame:0
          TX packets:12686 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:10500968 (10.0 MiB)  TX bytes:12513933 (11.9 MiB)

eth1      Link encap:Ethernet  HWaddr 00:01:DE:AD:BE:EF
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:82 errors:0 dropped:0 overruns:0 frame:0
          TX packets:82 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6220 (6.0 KiB)  TX bytes:6220 (6.0 KiB)

root@am64xx-evm:~#
```

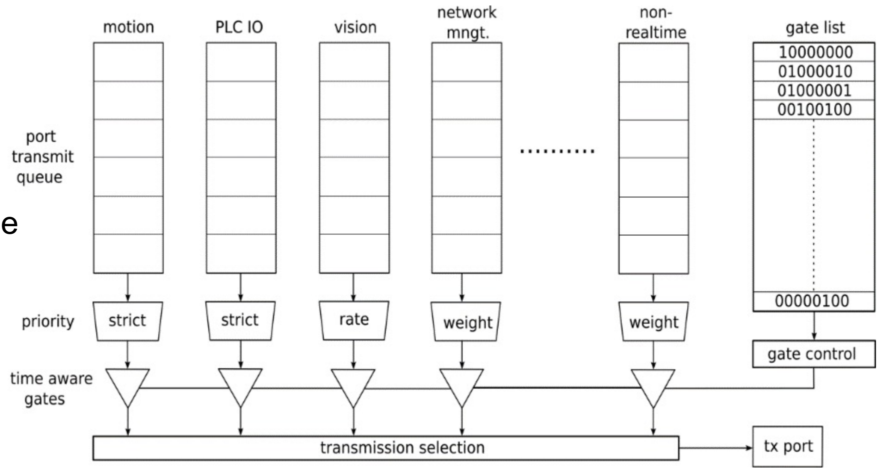
Development LAN, Linux Host

TSN Support on AM64x

- Time Sensitive Networking is an umbrella term for several optional Ethernet layer 2 features for achieving realtime performance
  - See white paper https://www.ti.com/lit/spry316 for additional background
  - Support is available in several Sitara and Jacinto processors
- The key features supported on Sitara AM64x Linux (iproute2 package) are:
  - **Integrated switching including cut-thru** (Cut-thru is not a IEEE802.1 standard yet)
    - Linux SW support and line topology demo (Sitara SDK 7.3.1)
  - **Timing over packet** (802.1AS-2011, IEEE1588, gPTP) with linuxptp (ptp4l)
  - **Preemption** (IET, 802.1Qbu/802.3br) with Linux packet scheduler (ethtool, tc qdisc)
  - **Time aware shaper** (EST, 802.1Qbv) with Linux packet scheduler (tc qdisc)
- Quality of service (QoS) including TSN works by preventing interference from lower priority traffic
  - It is the realtime protocol and application that must leverage the lack of interference to meet realtime

Enhancements to Scheduled Traffic (EST), Time Aware Shaper

- A transmit port feature typically synchronized across the local area network (LAN)
- Cyclic timeslots to allow realtime traffic to meet deadlines
  - E.g every 1000µs for 125µs only motion control traffic will be sent
- Uses 8 hardware queues per transmit port
  - Can be serialized with other QoS
- Typically require an network engineering tool to configure the LAN
  - With AM64x this would be software configuring the Linux packet scheduler at each transmit port
- Requires accurate PTP to align application endpoint and switch timing
- 128 entry gate control list, each entry contains:
  - Bitmask of open gates (queues) and duration
  - The sum of the times is the cycle time



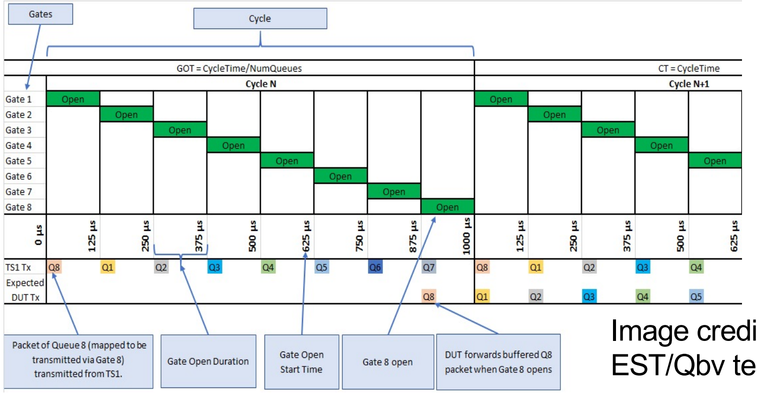
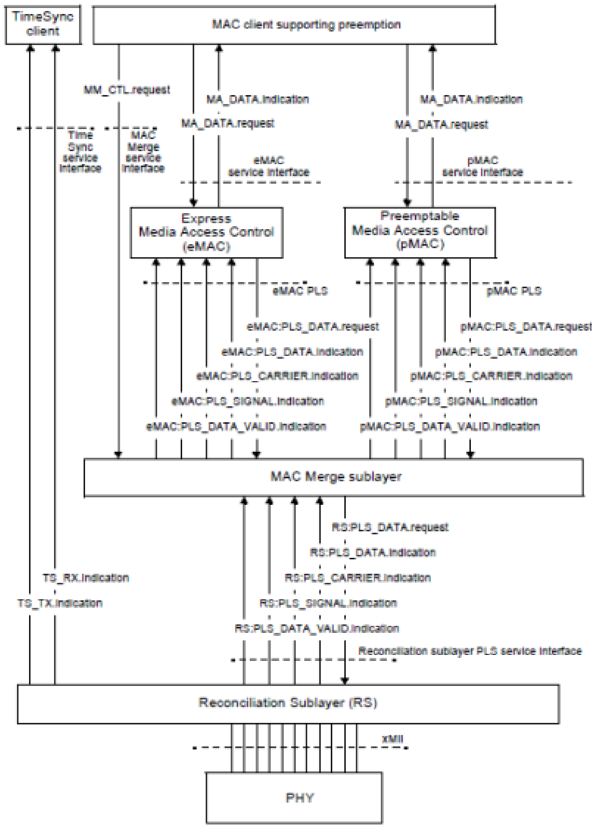


Image credit AVNU EST/Qbv test specification

Interspersing Express Traffic (IET), Preemption

- Port to port feature
- Splits MAC into two
  - Express traffic
  - Preemptable traffic
- 8 hardware queues per transmit port
  - Each queue can be express or preemptable
  - Can be serialized with other QoS
  - No nested preemption
- Does not require an network engineering tool to configure the LAN
  - LLDP messages to announce capability and minimum supported non-final fragment size
  - AM64x supports the optional further preamble level verify and respond sequence
- Does not require accurate PTP but only separates express traffic from all other traffic
- Cut-thru switching is only supported for express traffic

IEEE Std 802.3-2018, IEEE Standard for Ethernet
SECTION SEVEN



TimeSync client
MAC client supporting preemption
MM_CTL.request
MA_DATA.indication
MA_DATA.request
MA_DATA.indication
MA_DATA.request
Time Sync service interface
MAC Merge service interface
eMAC service interface
pMAC service interface
Express Media Access Control (eMAC)
Preemptable Media Access Control (pMAC)
eMAC PLS
pMAC PLS
eMAC:PLS_DATA.request
eMAC:PLS_DATA.indication
eMAC:PLS_CARRIER.indication
eMAC:PLS_SIGNAL.indication
eMAC:PLS_DATA_VALID.indication
pMAC:PLS_DATA.request
pMAC:PLS_DATA.indication
pMAC:PLS_CARRIER.indication
pMAC:PLS_SIGNAL.indication
pMAC:PLS_DATA_VALID.indication
MAC Merge sublayer
RS:PLS_DATA.request
RS:PLS_DATA.indication
RS:PLS_CARRIER.indication
RS:PLS_SIGNAL.indication
RS:PLS_DATA_VALID.indication
TS_RX.indication
TS_TX.indication
Reconciliation sublayer PLS service interface
Reconciliation Sublayer (RS)
xMII
PHY

Figure 99–2—MAC Merge sublayer service interfaces diagram

Linux Packet Scheduler and TSN

- *tc* (Traffic Classes) and *qdisc* (queueing discipline)
- Maps 16 socket (SKB) priorities to queues that can optionally be offloaded to HW
  - Software shaping is relevant orders of magnitude above typical TSN system
  - Credit Based Shaper (Ethernet AVB) has been supported for years
  - CPSW support for both EST and IET is supported in 5.4 kernel
- For more general description on the topic:
  - https://tsn.readthedocs.io/qdiscs.html

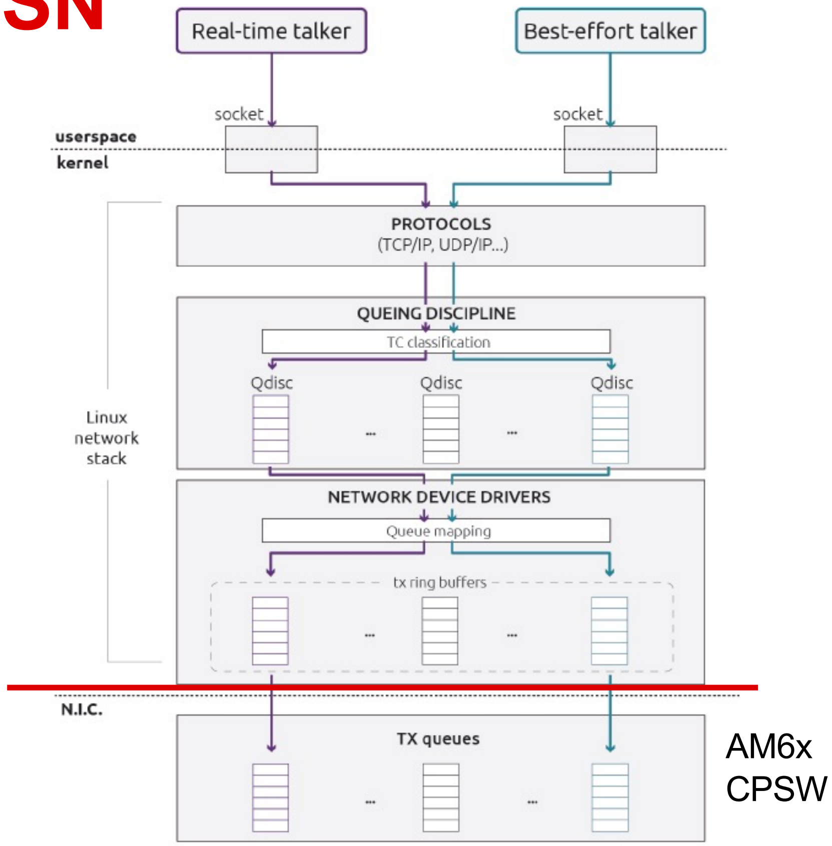
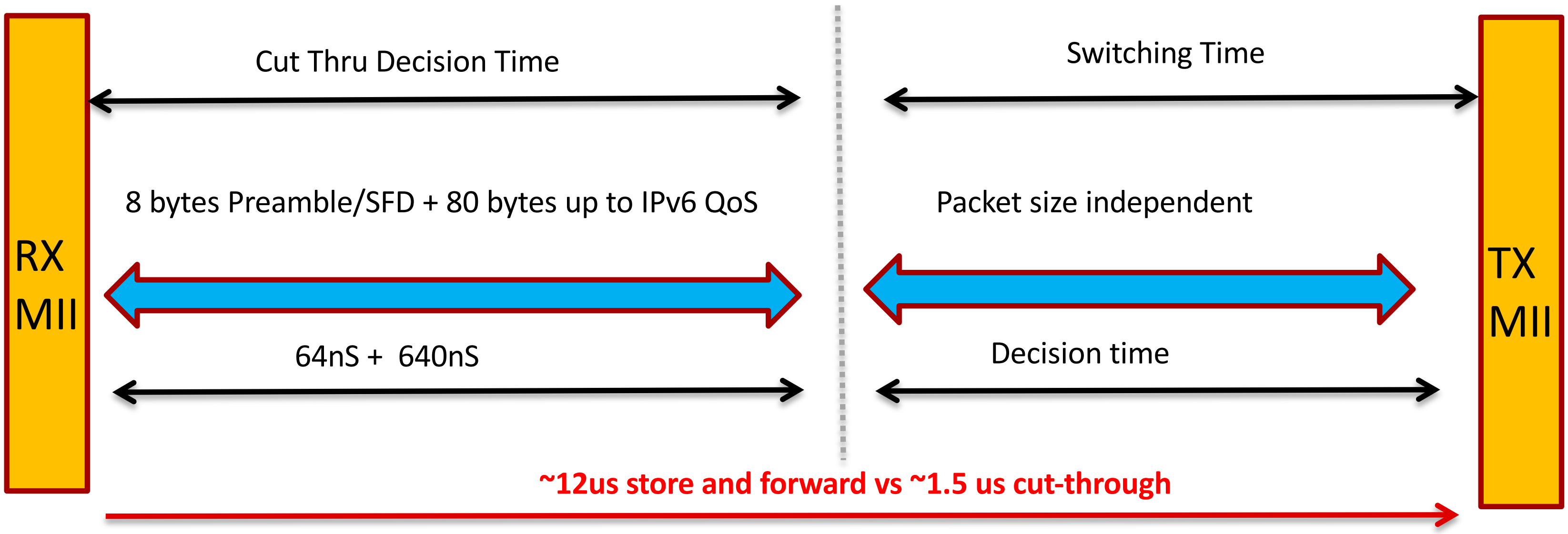


Fig. 1: Linux networking stack transmission path

https://arxiv.org/pdf/1808.10821.pdf

Cut-Through with AM6442 (CPSW3G)

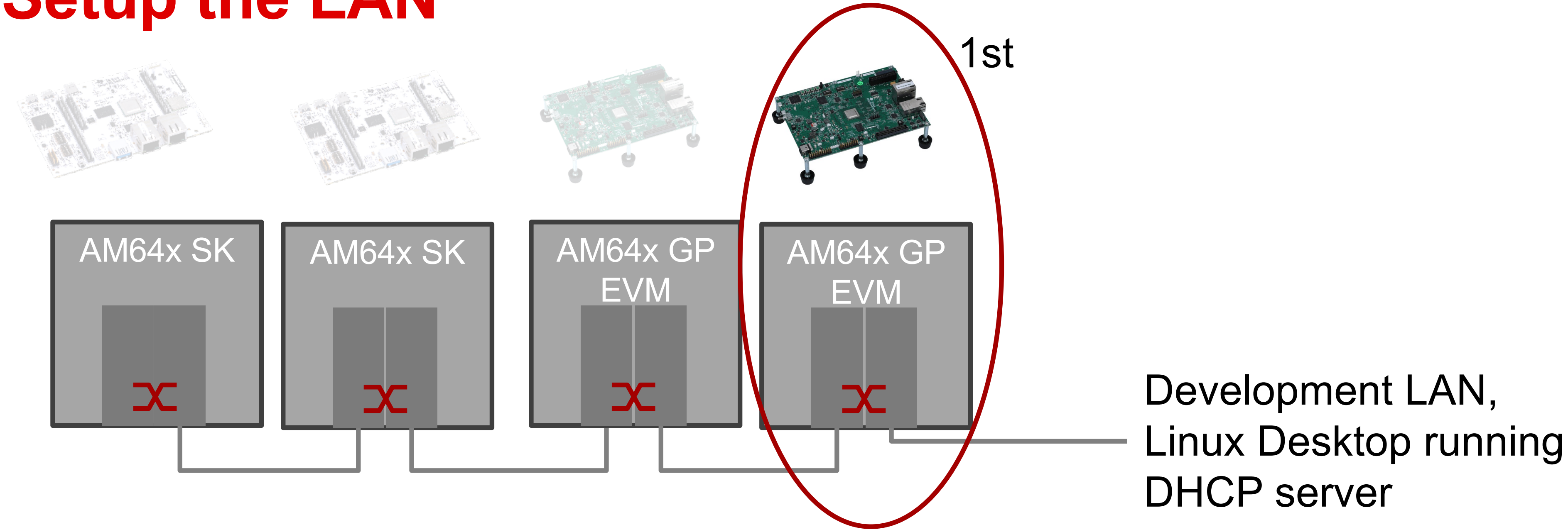


- Standard Ethernet Store and forward receives the entire frame, then forwards it
  - 1500 byte frame at 1Gbit/s takes 12us (1500 bytes times 8 ns per byte) to receive

- For Cut-Through decision is being taken after receiving 80 bytes. Lookup supports up to IPv6 QoS bits
- It takes 640 nanoseconds to receive 80 bytes at 1Gbit/s
- There is 8 byte preamble/SFD which takes 64 ns
- It takes further fixed processing to decide and start the transmission of packet.

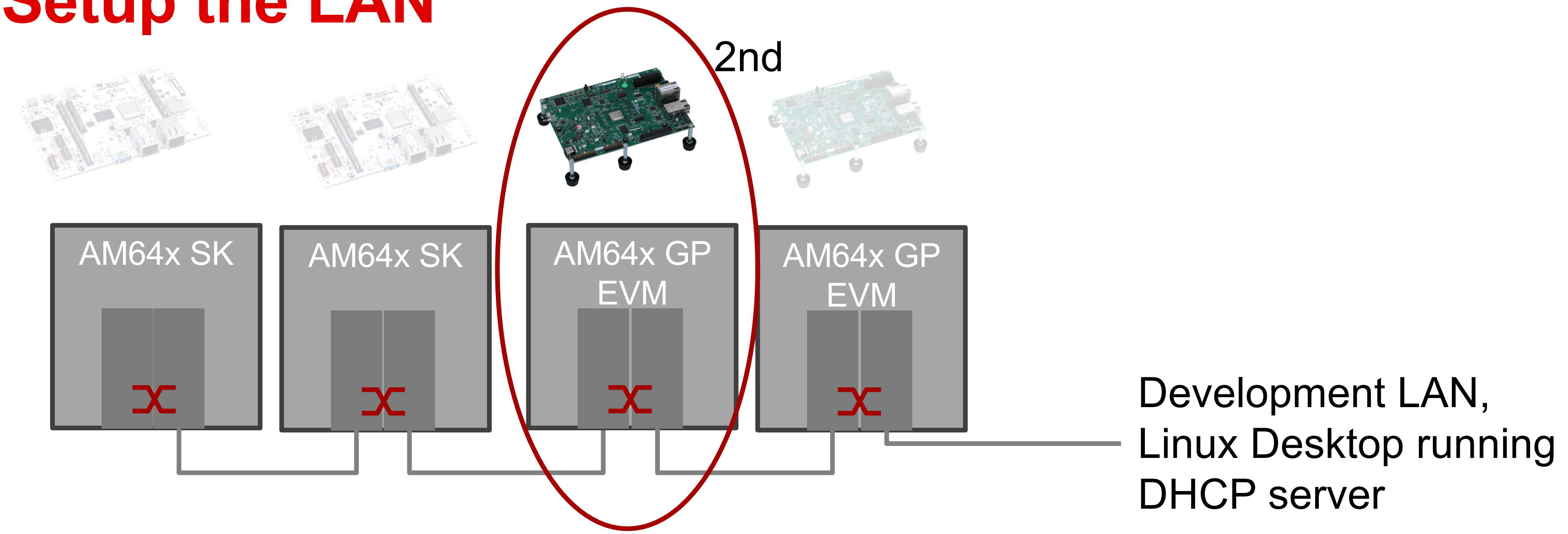
Questions

Setup the LAN



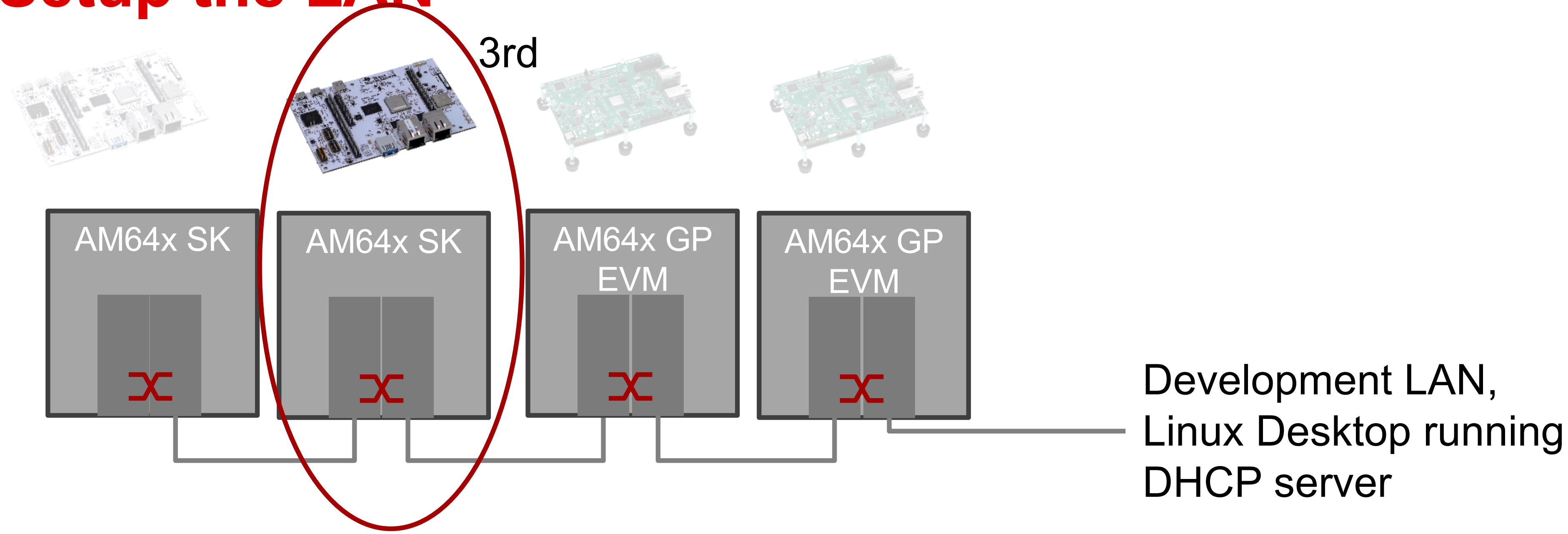
1. Connect USB console to each board (see AM64x SK Quick Start Guide)
2. Starting with the board connected to your Linux machine (on the right above)
   1. By default the AM64x EVM and SK come up in dual MAC mode
   2. Configure as cut through switch (later slide)

Setup the LAN



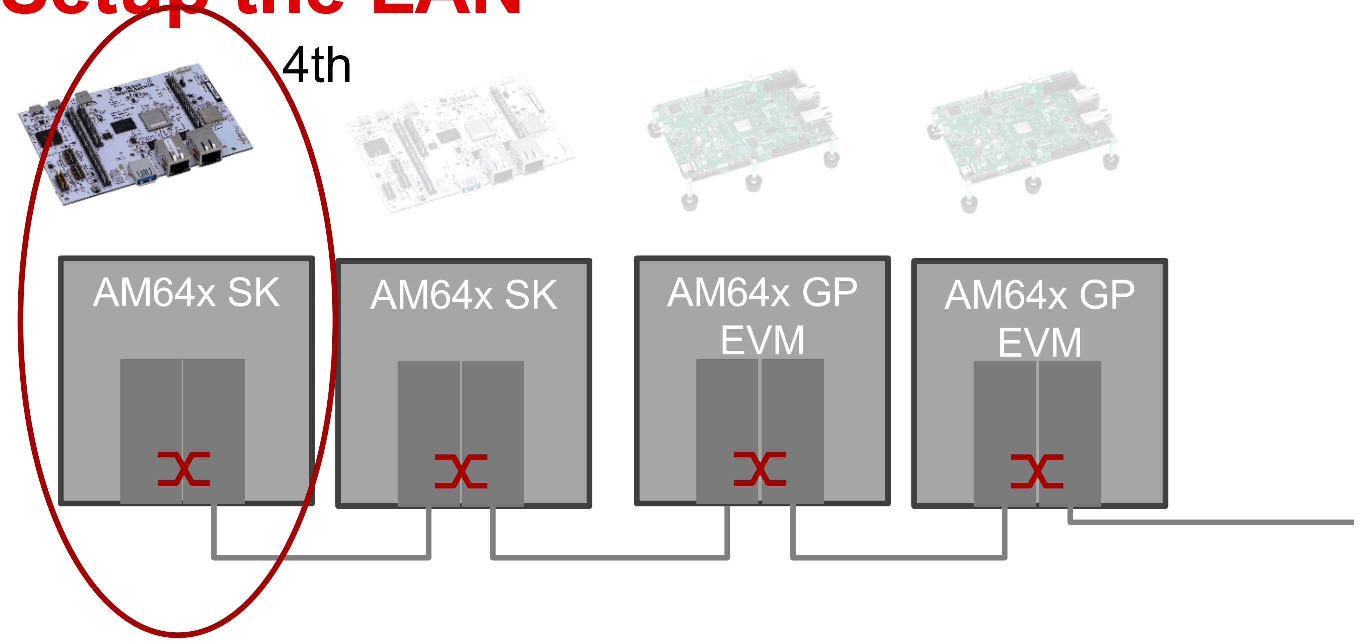
1. Connect USB console to each board (see AM64x SK Quick Start Guide)
2. Starting with the board connected to your Linux machine (on the right above)
   1. By default the AM64x EVM and SK come up in dual MAC mode
   2. Configure as cut through switch (later slide)

Setup the LAN



1. Connect USB console to each board (see AM64x SK Quick Start Guide)
2. Starting with the board connected to your Linux machine (on the right above)
    1. By default the AM64x EVM and SK come up in dual MAC mode
    2. Configure as cut through switch (later slide)

Setup the LAN



Development LAN, Linux Desktop running DHCP server

1. Connect USB console to each board (see AM64x SK Quick Start Guide)
2. Starting with the board connected to your Linux machine (on the right above)
   1. By default the AM64x EVM and SK come up in dual MAC mode
   2. Configure as cut through switch (later slide)

Script to Configure each AM64x for cut-through Switch with Preemption

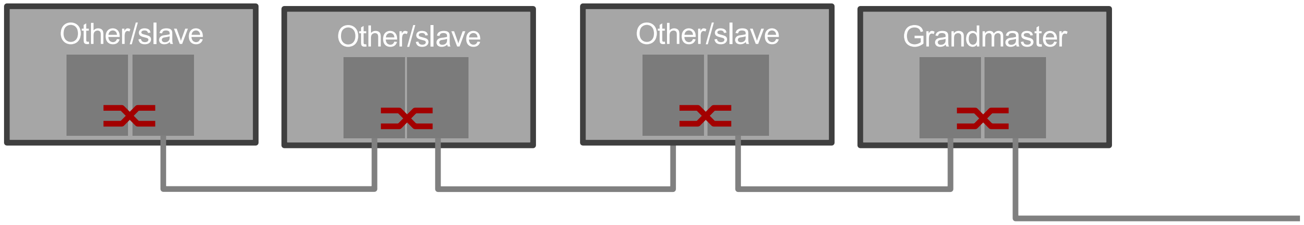
```
echo "Script to initialize AM64x Ethernet ports (CPSW3G) as hardware based 802.1Q cut-through";
echo "bridge using ip and devlink commands.";
echo
echo "setting cut-thru and preemption for VLAN PCP 3 , bitmask 8 0b00001000";
echo
#take the interfaces down, by default dual MAC
ip link set dev eth0 down
ip link set dev eth1 down
# two queues one express, one non-express, and SMD level verify on
ethtool -L eth0 tx 2
ethtool -L eth1 tx 2
ethtool --set-priv-flags eth0 p0-rx-ptype-rrobin off
ethtool --set-priv-flags eth0 iet-frame-preemption on
ethtool --set-priv-flags eth0 iet-mac-verify on
ethtool --set-priv-flags eth1 p0-rx-ptype-rrobin off
ethtool --set-priv-flags eth1 iet-frame-preemption on
ethtool --set-priv-flags eth1 iet-mac-verify on
#tell the kernel hw switch to turn on switching
devlink dev param set platform/8000000.ethernet name switch_mode value true cmode runtime
#setup Q7 of 8 queues to have cut-thru on
echo 8 > /sys/kernel/debug/8000000.ethernet/Port1/cut_thru_rx_pri_mask
echo 8 > /sys/kernel/debug/8000000.ethernet/Port1/cut_thru_tx_pri_mask
echo 8 > /sys/kernel/debug/8000000.ethernet/Port2/cut_thru_rx_pri_mask
echo 8 > /sys/kernel/debug/8000000.ethernet/Port2/cut_thru_tx_pri_mask
ethtool   --set-priv-flags eth0 cut-thru  on
ethtool   --set-priv-flags eth1 cut-thru  on
```

Script to Configure each AM64x as a Switch

```
#setup a host port called br0
ip link add name br0 type bridge
ip link set dev br0 type bridge ageing_time 1000
#bring up th external Ethernet MACs, sleep is there to allow the link negotiation to go through
ip link set dev eth0 up
sleep 1
ip link set dev eth1 up
sleep 1
#connect the external MACs to the host port
ip link set dev eth0 master br0
ip link set dev eth1 master br0
#tell that the host port is part of the bridge
ip link set dev br0 type bridge stp_state 1
#bring the host port up
ip link set dev br0 up
#CPSW switch comes up in vlan aware mode, with ports blocked by default
#vlan id 1 is used for untagged traffic with VLANs, this unblock the untagged traffic to br0 por
t
bridge vlan add dev br0 vid 1 pvid untagged self
#echo spanning tree for the switch
brctl showstp br0
#get an IP address for the host port using DHCP
udhcpc -i br0
```

IEEE 802.1AS-2011 using ptp4l

- ptp4l with ptp relay at the AM64x's doing the switching
  - converges to single digit accuracy with outliers at ~10ns
  - to get the Linux clock to use this ptp time run from the command line, or set it up as a service.
- On the grandmaster (options are autoconfig, system clock, and allow system clock to be source, --transportSpecific=1 means 802.1AS domain in 1588):
  - `date -s "27 MAY 2021 14:27:10"`
  - `phc2sys -a -rr --transportSpecific=1 &`
- At the others two (options are autoconfig, system clock):
  - `phc2sys -a -r --transportSpecific=1 &`

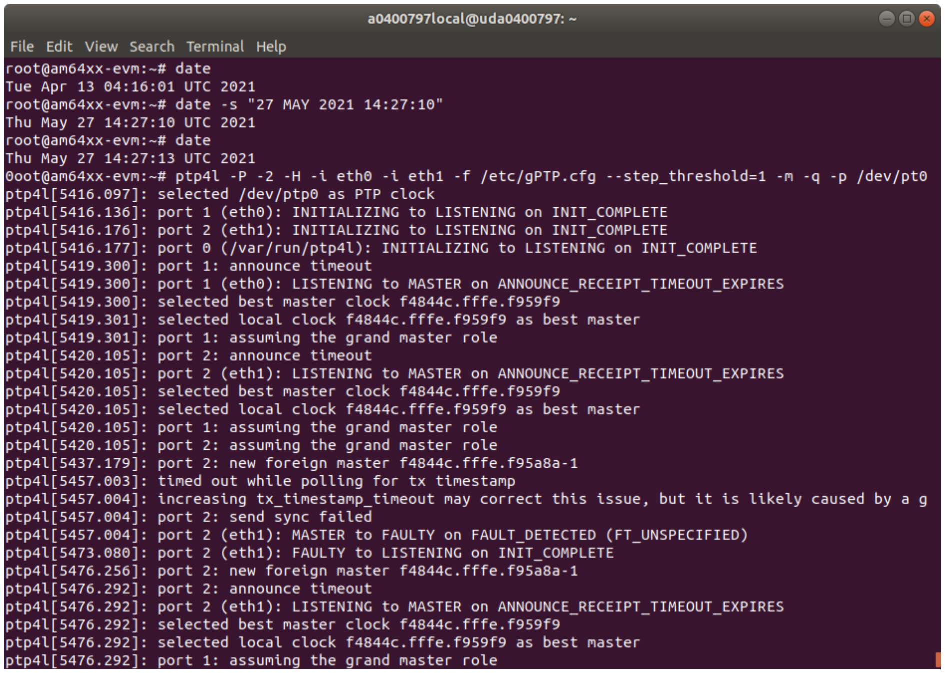


Start ptp4l

- Grandmaster AM64x first then others:
  - `ptp4l -P -2 -H -i eth0 -i eth1 -f /etc/gPTP.cfg --step_threshold=1 -m -q -p /dev/ptp0`
    - P means peer to peer (as opposed to end to end)
    - 2 means Ethernet level IEEE1588 (not IPv4 or IPv6)
    - H means use HW timestamps
    - i is interface
    - m is print to stdout
    - q is don't print to system log
    - p is what clock on the AM6x to use
    - gPTP.cfg is a file from https://github.com/richardcochran/linuxptp/blob/master/configs/gPTP.cfg
      - Two lines added based on the PHY datasheet on the EVM *ingressLatency 88, egressLatency 288*
    - step_threshold=1 is for faster convergence on jumps
  - On the other boards you can also check with command *date* to see Linux time has adjusted
- See https://tsn.readthedocs.io/timesync.html for more details

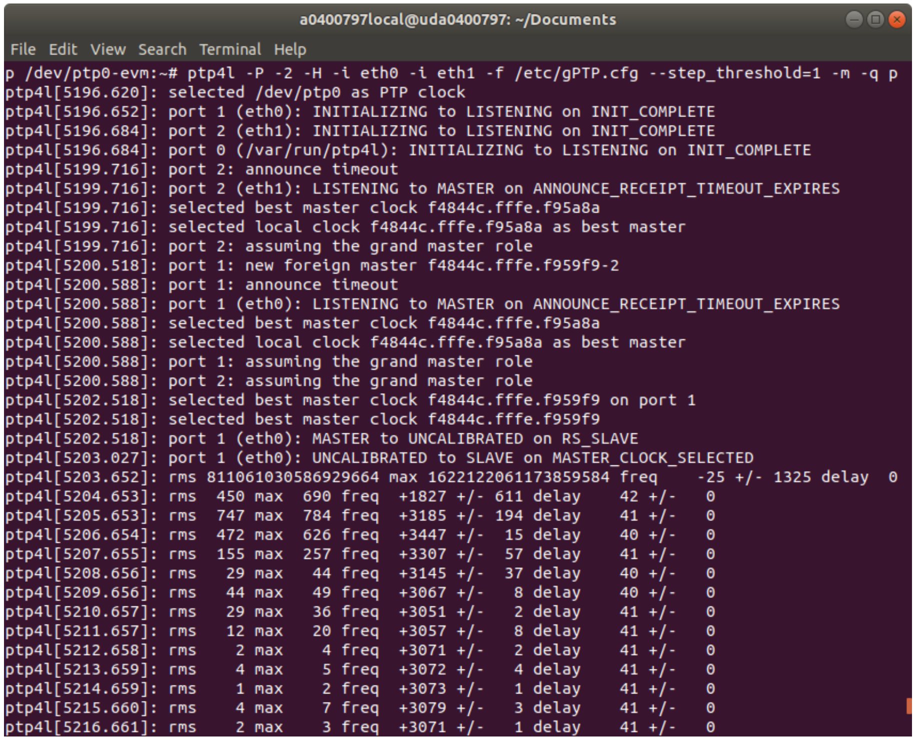
IEEE802.1AS-2011 running

Grandmaster


a0400797local@uda0400797: ~
File Edit View Search Terminal Help

```
root@am64xx-evm:~# date
Tue Apr 13 04:16:01 UTC 2021
root@am64xx-evm:~# date -s "27 MAY 2021 14:27:10"
Thu May 27 14:27:10 UTC 2021
root@am64xx-evm:~# date
Thu May 27 14:27:13 UTC 2021
0oot@am64xx-evm:~# ptp4l -P -2 -H -i eth0 -i eth1 -f /etc/gPTP.cfg --step_threshold=1 -m -q -p /dev/pt0
ptp4l[5416.097]: selected /dev/ptp0 as PTP clock
ptp4l[5416.136]: port 1 (eth0): INITIALIZING to LISTENING on INIT_COMPLETE
ptp4l[5416.176]: port 2 (eth1): INITIALIZING to LISTENING on INIT_COMPLETE
ptp4l[5416.177]: port 0 (/var/run/ptp4l): INITIALIZING to LISTENING on INIT_COMPLETE
ptp4l[5419.300]: port 1: announce timeout
ptp4l[5419.300]: port 1 (eth0): LISTENING to MASTER on ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES
ptp4l[5419.300]: selected best master clock f4844c.fffe.f959f9
ptp4l[5419.301]: selected local clock f4844c.fffe.f959f9 as best master
ptp4l[5419.301]: port 1: assuming the grand master role
ptp4l[5420.105]: port 2: announce timeout
ptp4l[5420.105]: port 2 (eth1): LISTENING to MASTER on ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES
ptp4l[5420.105]: selected best master clock f4844c.fffe.f959f9
ptp4l[5420.105]: selected local clock f4844c.fffe.f959f9 as best master
ptp4l[5420.105]: port 1: assuming the grand master role
ptp4l[5420.105]: port 2: assuming the grand master role
ptp4l[5437.179]: port 2: new foreign master f4844c.fffe.f95a8a-1
ptp4l[5457.003]: timed out while polling for tx timestamp
ptp4l[5457.004]: increasing tx_timestamp_timeout may correct this issue, but it is likely caused by a g
ptp4l[5457.004]: port 2: send sync failed
ptp4l[5457.004]: port 2 (eth1): MASTER to FAULTY on FAULT_DETECTED (FT_UNSPECIFIED)
ptp4l[5473.080]: port 2 (eth1): FAULTY to LISTENING on INIT_COMPLETE
ptp4l[5476.256]: port 2: new foreign master f4844c.fffe.f95a8a-1
ptp4l[5476.292]: port 2: announce timeout
ptp4l[5476.292]: port 2 (eth1): LISTENING to MASTER on ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES
ptp4l[5476.292]: selected best master clock f4844c.fffe.f959f9
ptp4l[5476.292]: selected local clock f4844c.fffe.f959f9 as best master
ptp4l[5476.292]: port 1: assuming the grand master role
```

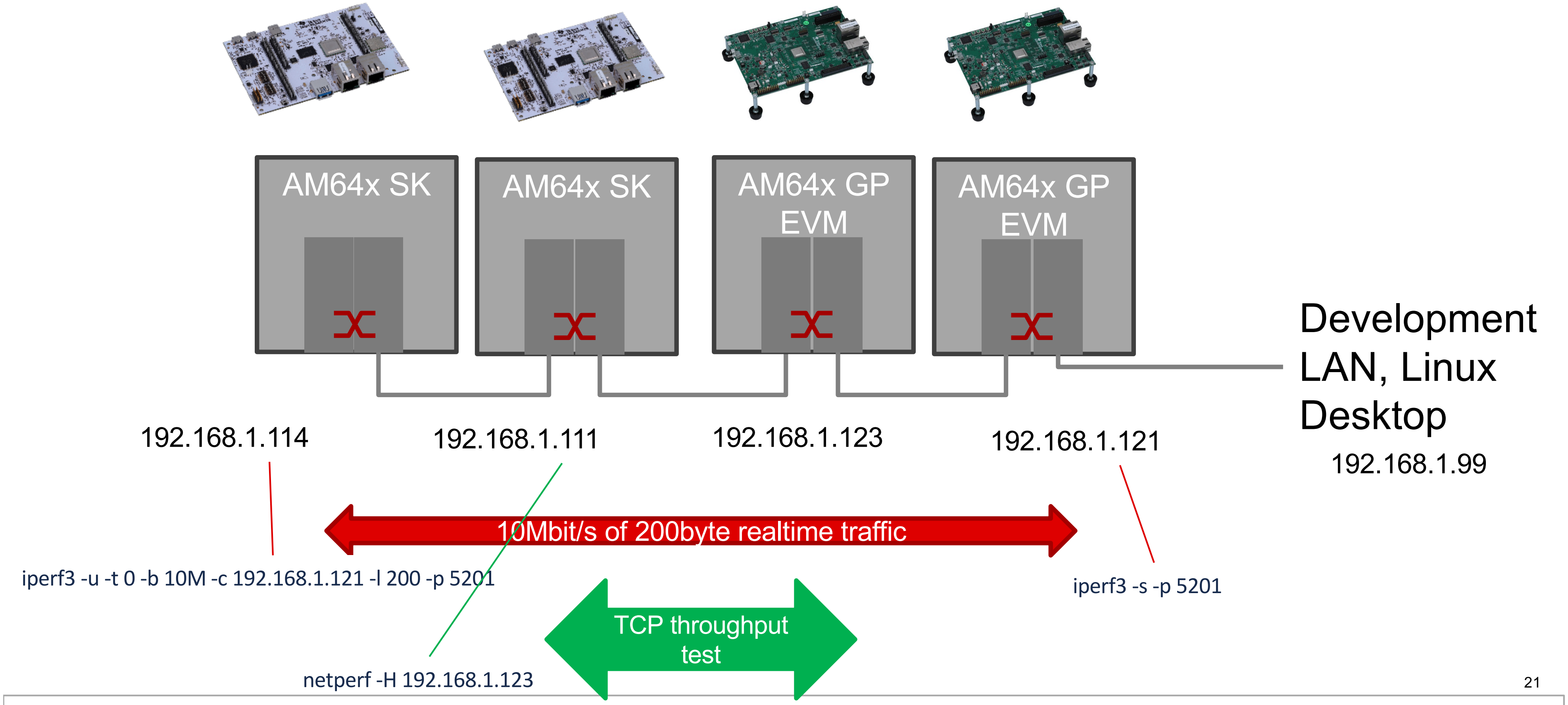
Others


a0400797local@uda0400797: ~/Documents
File Edit View Search Terminal Help

```
p /dev/ptp0-evm:~# ptp4l -P -2 -H -i eth0 -i eth1 -f /etc/gPTP.cfg --step_threshold=1 -m -q p
ptp4l[5196.620]: selected /dev/ptp0 as PTP clock
ptp4l[5196.652]: port 1 (eth0): INITIALIZING to LISTENING on INIT_COMPLETE
ptp4l[5196.684]: port 2 (eth1): INITIALIZING to LISTENING on INIT_COMPLETE
ptp4l[5196.684]: port 0 (/var/run/ptp4l): INITIALIZING to LISTENING on INIT_COMPLETE
ptp4l[5199.716]: port 2: announce timeout
ptp4l[5199.716]: port 2 (eth1): LISTENING to MASTER on ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES
ptp4l[5199.716]: selected best master clock f4844c.fffe.f95a8a
ptp4l[5199.716]: selected local clock f4844c.fffe.f95a8a as best master
ptp4l[5199.716]: port 2: assuming the grand master role
ptp4l[5200.518]: port 1: new foreign master f4844c.fffe.f959f9-2
ptp4l[5200.588]: port 1: announce timeout
ptp4l[5200.588]: port 1 (eth0): LISTENING to MASTER on ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES
ptp4l[5200.588]: selected best master clock f4844c.fffe.f95a8a
ptp4l[5200.588]: selected local clock f4844c.fffe.f95a8a as best master
ptp4l[5200.588]: port 1: assuming the grand master role
ptp4l[5200.588]: port 2: assuming the grand master role
ptp4l[5202.518]: selected best master clock f4844c.fffe.f959f9 on port 1
ptp4l[5202.518]: selected best master clock f4844c.fffe.f959f9
ptp4l[5202.518]: port 1 (eth0): MASTER to UNCALIBRATED on RS_SLAVE
ptp4l[5203.027]: port 1 (eth0): UNCALIBRATED to SLAVE on MASTER_CLOCK_SELECTED
ptp4l[5203.652]: rms 811061030586929664 max 1622122061173859584 freq    -25 +/- 1325 delay  0
ptp4l[5204.653]: rms  450 max  690 freq  +1827 +/- 611 delay    42 +/-   0
ptp4l[5205.653]: rms  747 max  784 freq  +3185 +/- 194 delay    41 +/-   0
ptp4l[5206.654]: rms  472 max  626 freq  +3447 +/-  15 delay    40 +/-   0
ptp4l[5207.655]: rms  155 max  257 freq  +3307 +/-  57 delay    41 +/-   0
ptp4l[5208.656]: rms   29 max   44 freq  +3145 +/-  37 delay    40 +/-   0
ptp4l[5209.656]: rms   44 max   49 freq  +3067 +/-   8 delay    40 +/-   0
ptp4l[5210.657]: rms   29 max   36 freq  +3051 +/-   2 delay    41 +/-   0
ptp4l[5211.657]: rms   12 max   20 freq  +3057 +/-   8 delay    41 +/-   0
ptp4l[5212.658]: rms    2 max    4 freq  +3071 +/-   2 delay    41 +/-   0
ptp4l[5213.659]: rms    4 max    5 freq  +3072 +/-   4 delay    41 +/-   0
ptp4l[5214.659]: rms    1 max    2 freq  +3073 +/-   1 delay    41 +/-   0
ptp4l[5215.660]: rms    4 max    7 freq  +3079 +/-   3 delay    41 +/-   0
ptp4l[5216.661]: rms    2 max    3 freq  +3071 +/-   1 delay    41 +/-   0
```

Questions

Realtime Traffic without QoS



AM64x SK

AM64x SK

AM64x GP EVM

AM64x GP EVM

Development LAN, Linux Desktop

192.168.1.114

192.168.1.111

192.168.1.123

192.168.1.121

192.168.1.99

10Mbit/s of 200byte realtime traffic

TCP throughput test

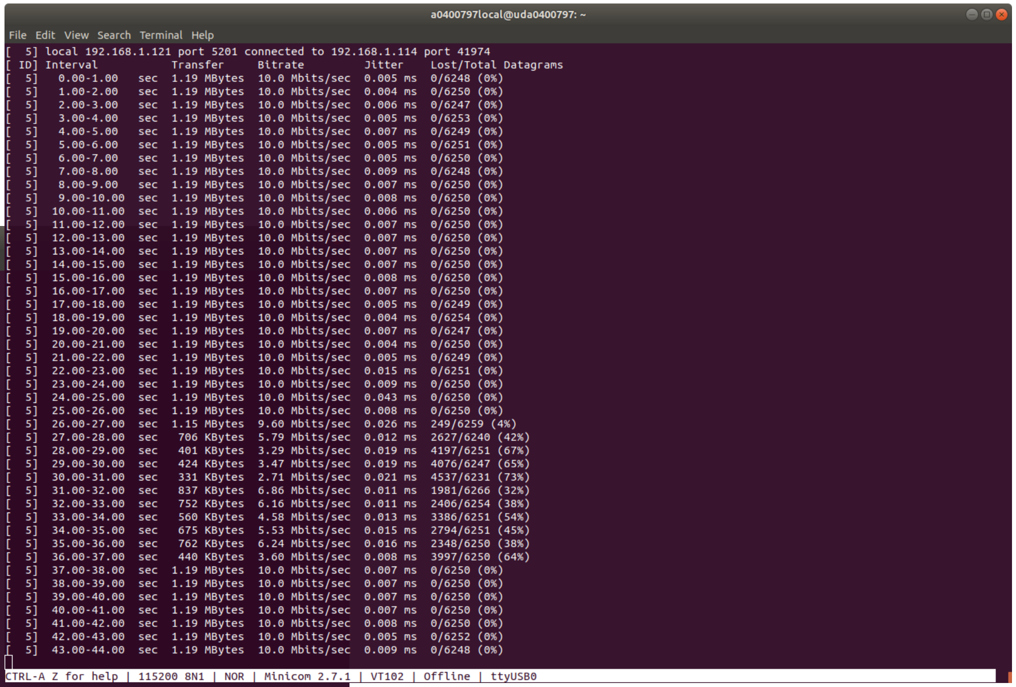
```
iperf3 -u -t 0 -b 10M -c 192.168.1.121 -l 200 -p 5201
```

```
iperf3 -s -p 5201
```

```
netperf -H 192.168.1.123
```

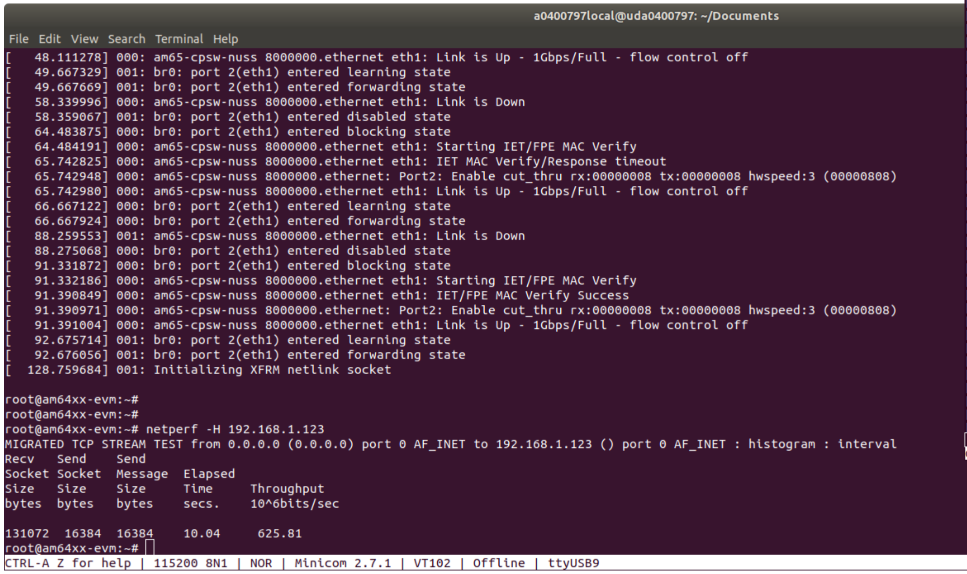
Realtime Traffic without QoS

Realtime (iperf3)


a0400797local@uda0400797: ~
File Edit View Search Terminal Help

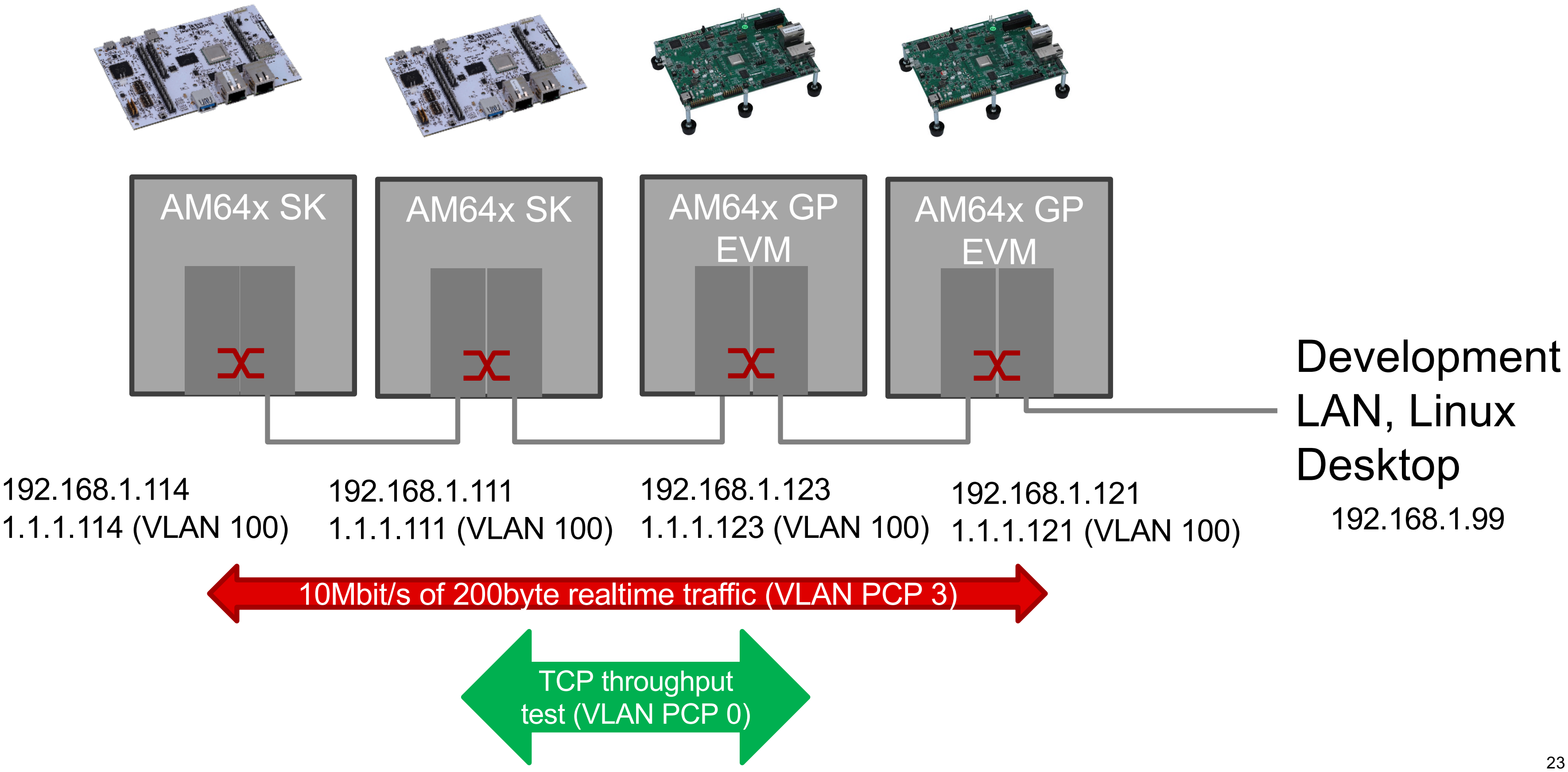
```
[  5] local 192.168.1.121 port 5201 connected to 192.168.1.114 port 41974
[ ID] Interval           Transfer     Bitrate         Jitter    Lost/Total Datagrams
[  5]   0.00-1.00   sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6248 (0%)
[  5]   1.00-2.00   sec  1.19 MBytes  10.0 Mbits/sec  0.004 ms  0/6250 (0%)
[  5]   2.00-3.00   sec  1.19 MBytes  10.0 Mbits/sec  0.006 ms  0/6247 (0%)
[  5]   3.00-4.00   sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6253 (0%)
[  5]   4.00-5.00   sec  1.19 MBytes  10.0 Mbits/sec  0.007 ms  0/6249 (0%)
[  5]   5.00-6.00   sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6251 (0%)
[  5]   6.00-7.00   sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6250 (0%)
[  5]   7.00-8.00   sec  1.19 MBytes  10.0 Mbits/sec  0.009 ms  0/6248 (0%)
[  5]   8.00-9.00   sec  1.19 MBytes  10.0 Mbits/sec  0.007 ms  0/6250 (0%)
[  5]   9.00-10.00  sec  1.19 MBytes  10.0 Mbits/sec  0.008 ms  0/6250 (0%)
[  5]  10.00-11.00  sec  1.19 MBytes  10.0 Mbits/sec  0.006 ms  0/6250 (0%)
[  5]  11.00-12.00  sec  1.19 MBytes  10.0 Mbits/sec  0.007 ms  0/6250 (0%)
[  5]  12.00-13.00  sec  1.19 MBytes  10.0 Mbits/sec  0.007 ms  0/6250 (0%)
[  5]  13.00-14.00  sec  1.19 MBytes  10.0 Mbits/sec  0.007 ms  0/6250 (0%)
[  5]  14.00-15.00  sec  1.19 MBytes  10.0 Mbits/sec  0.007 ms  0/6250 (0%)
[  5]  15.00-16.00  sec  1.19 MBytes  10.0 Mbits/sec  0.008 ms  0/6250 (0%)
[  5]  16.00-17.00  sec  1.19 MBytes  10.0 Mbits/sec  0.007 ms  0/6250 (0%)
[  5]  17.00-18.00  sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6249 (0%)
[  5]  18.00-19.00  sec  1.19 MBytes  10.0 Mbits/sec  0.004 ms  0/6254 (0%)
[  5]  19.00-20.00  sec  1.19 MBytes  10.0 Mbits/sec  0.007 ms  0/6247 (0%)
[  5]  20.00-21.00  sec  1.19 MBytes  10.0 Mbits/sec  0.004 ms  0/6250 (0%)
[  5]  21.00-22.00  sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6249 (0%)
[  5]  22.00-23.00  sec  1.19 MBytes  10.0 Mbits/sec  0.015 ms  0/6251 (0%)
[  5]  23.00-24.00  sec  1.19 MBytes  10.0 Mbits/sec  0.009 ms  0/6250 (0%)
[  5]  24.00-25.00  sec  1.19 MBytes  10.0 Mbits/sec  0.043 ms  0/6250 (0%)
[  5]  25.00-26.00  sec  1.19 MBytes  10.0 Mbits/sec  0.008 ms  0/6250 (0%)
[  5]  26.00-27.00  sec  1.15 MBytes  9.60 Mbits/sec  0.026 ms  249/6259 (4%)
[  5]  27.00-28.00  sec   706 KBytes  5.79 Mbits/sec  0.012 ms  2627/6240 (42%)
[  5]  28.00-29.00  sec   401 KBytes  3.29 Mbits/sec  0.019 ms  4197/6251 (67%)
[  5]  29.00-30.00  sec   424 KBytes  3.47 Mbits/sec  0.019 ms  4076/6247 (65%)
[  5]  30.00-31.00  sec   331 KBytes  2.71 Mbits/sec  0.021 ms  4537/6231 (73%)
[  5]  31.00-32.00  sec   837 KBytes  6.86 Mbits/sec  0.011 ms  1981/6266 (32%)
[  5]  32.00-33.00  sec   752 KBytes  6.16 Mbits/sec  0.011 ms  2406/6254 (38%)
[  5]  33.00-34.00  sec   560 KBytes  4.58 Mbits/sec  0.013 ms  3386/6251 (54%)
[  5]  34.00-35.00  sec   675 KBytes  5.53 Mbits/sec  0.015 ms  2794/6251 (45%)
[  5]  35.00-36.00  sec   762 KBytes  6.24 Mbits/sec  0.016 ms  2348/6250 (38%)
[  5]  36.00-37.00  sec   440 KBytes  3.60 Mbits/sec  0.008 ms  3997/6250 (64%)
[  5]  37.00-38.00  sec  1.19 MBytes  10.0 Mbits/sec  0.007 ms  0/6250 (0%)
[  5]  38.00-39.00  sec  1.19 MBytes  10.0 Mbits/sec  0.007 ms  0/6250 (0%)
[  5]  39.00-40.00  sec  1.19 MBytes  10.0 Mbits/sec  0.007 ms  0/6250 (0%)
[  5]  40.00-41.00  sec  1.19 MBytes  10.0 Mbits/sec  0.007 ms  0/6250 (0%)
[  5]  41.00-42.00  sec  1.19 MBytes  10.0 Mbits/sec  0.008 ms  0/6250 (0%)
[  5]  42.00-43.00  sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6252 (0%)
[  5]  43.00-44.00  sec  1.19 MBytes  10.0 Mbits/sec  0.009 ms  0/6248 (0%)

CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | ttyUSB0
```

Background throughput test (netperf)


a0400797local@uda0400797: ~/Documents
File Edit View Search Terminal Help

```
[   48.111278] 000: am65-cpsw-nuss 8000000.ethernet eth1: Link is Up - 1Gbps/Full - flow control off
[   49.667329] 001: br0: port 2(eth1) entered learning state
[   49.667669] 001: br0: port 2(eth1) entered forwarding state
[   58.339996] 000: am65-cpsw-nuss 8000000.ethernet eth1: Link is Down
[   58.359067] 001: br0: port 2(eth1) entered disabled state
[   64.483875] 000: br0: port 2(eth1) entered blocking state
[   64.484191] 000: am65-cpsw-nuss 8000000.ethernet eth1: Starting IET/FPE MAC Verify
[   65.742825] 000: am65-cpsw-nuss 8000000.ethernet eth1: IET MAC Verify/Response timeout
[   65.742948] 000: am65-cpsw-nuss 8000000.ethernet: Port2: Enable cut_thru rx:00000008 tx:00000008 hwspeed:3 (00000808)
[   65.742980] 000: am65-cpsw-nuss 8000000.ethernet eth1: Link is Up - 1Gbps/Full - flow control off
[   66.667122] 000: br0: port 2(eth1) entered learning state
[   66.667924] 000: br0: port 2(eth1) entered forwarding state
[   88.259553] 001: am65-cpsw-nuss 8000000.ethernet eth1: Link is Down
[   88.275068] 000: br0: port 2(eth1) entered disabled state
[   91.331872] 000: br0: port 2(eth1) entered blocking state
[   91.332186] 000: am65-cpsw-nuss 8000000.ethernet eth1: Starting IET/FPE MAC Verify
[   91.390849] 000: am65-cpsw-nuss 8000000.ethernet eth1: IET/FPE MAC Verify Success
[   91.390971] 000: am65-cpsw-nuss 8000000.ethernet: Port2: Enable cut_thru rx:00000008 tx:00000008 hwspeed:3 (00000808)
[   91.391004] 000: am65-cpsw-nuss 8000000.ethernet eth1: Link is Up - 1Gbps/Full - flow control off
[   92.675714] 001: br0: port 2(eth1) entered learning state
[   92.676056] 001: br0: port 2(eth1) entered forwarding state
[  128.759684] 001: Initializing XFRM netlink socket

root@am64xx-evm:~#
root@am64xx-evm:~#
root@am64xx-evm:~# netperf -H 192.168.1.123
MIGRATED TCP STREAM TEST from 0.0.0.0 (0.0.0.0) port 0 AF_INET to 192.168.1.123 () port 0 AF_INET : histogram : interval
Recv   Send    Send
Socket Socket  Message  Elapsed
Size   Size    Size     Time     Throughput
bytes  bytes   bytes    secs.    10^6bits/sec

131072  16384  16384    10.04     625.81
root@am64xx-evm:~#
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | ttyUSB9
```

Realtime Traffic with Preemption and Cut-through



AM64x SK

AM64x SK

AM64x GP EVM

AM64x GP EVM

Development LAN, Linux Desktop
192.168.1.99

192.168.1.114
1.1.1.114 (VLAN 100)

192.168.1.111
1.1.1.111 (VLAN 100)

192.168.1.123
1.1.1.123 (VLAN 100)

192.168.1.121
1.1.1.121 (VLAN 100)

10Mbit/s of 200byte realtime traffic (VLAN PCP 3)

TCP throughput test (VLAN PCP 0)

Use VLAN priority as TSN Stream Identifier

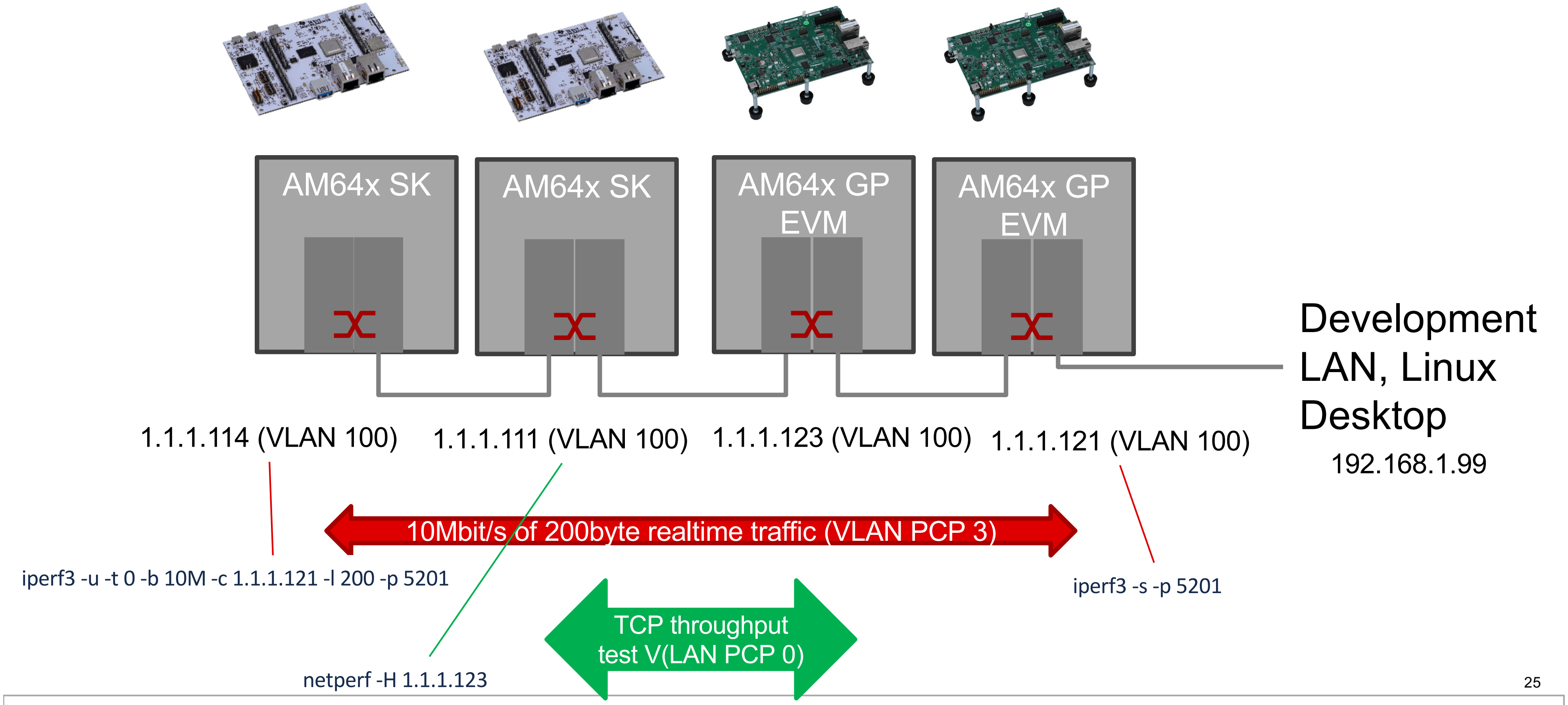
- Using VLAN priority (PCP) is perhaps the simplest usage of TSN
- Tools like iperf3 and netperf will use default priority (SO_PRIORITY) of 0
  - Map the devices with realtime so default socket uses VLAN PCP=3 which we made express and cut-through
  - In the middle two leave default as VLAN PCP=0
- Create a VLAN with ID 100 across the 4 devices

```
# middle two all traffic uses default
ip link add link br0 name br0.100 type vlan id 100
sleep 1
bridge vlan add dev br0 vid 100 self
bridge vlan add dev eth0 vid 100
bridge vlan add dev eth1 vid 100
ip link set dev br0.100 up
```

```
# endpoint two
ip link add link br0 name br0.100 type vlan id 100 egress-qos-map 0:3
sleep 1
bridge vlan add dev br0 vid 100 self
bridge vlan add dev eth0 vid 100
bridge vlan add dev eth1 vid 100
ip link set dev br0.100 up
```

- Give each VLAN device static IP address 1.1.1.x (e.g. where x is the same as the DHCP address)
  - ip addr add 1.1.1.114/24 dev br0.100
- In a real realtime application (not iperf3) you would use SO_PRIORITY other than default and appropriate mapping to TSN stream such as VLAN priority

Realtime Traffic with Cut-through and Preemeption



AM64x SK

AM64x SK

AM64x GP EVM

AM64x GP EVM

Development LAN, Linux Desktop

192.168.1.99

1.1.1.114 (VLAN 100)

1.1.1.111 (VLAN 100)

1.1.1.123 (VLAN 100)

1.1.1.121 (VLAN 100)

10Mbit/s of 200byte realtime traffic (VLAN PCP 3)

```
iperf3 -u -t 0 -b 10M -c 1.1.1.121 -l 200 -p 5201
```

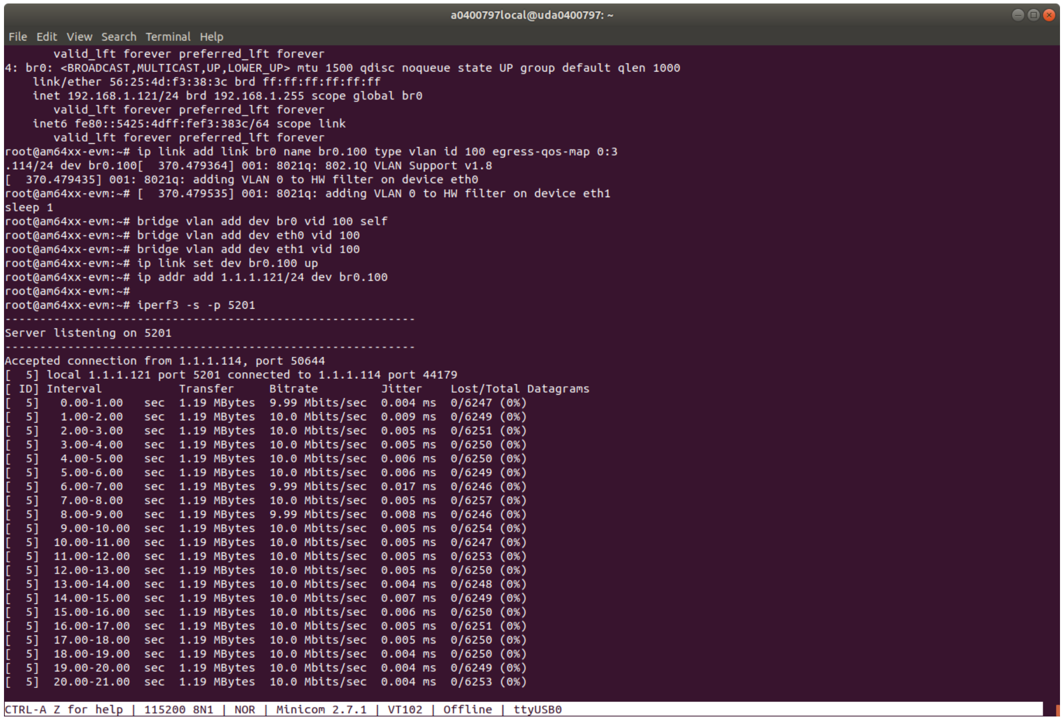
```
iperf3 -s -p 5201
```

TCP throughput test V(LAN PCP 0)

```
netperf -H 1.1.1.123
```

Realtime Traffic with QoS

Realtime (iperf3) using VLAN PCP=3 is unaffected by other traffic


a0400797local@uda0400797: ~
File Edit View Search Terminal Help

```
       valid_lft forever preferred_lft forever
4: br0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 56:25:4d:f3:38:3c brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.121/24 brd 192.168.1.255 scope global br0
       valid_lft forever preferred_lft forever
    inet6 fe80::5425:4dff:fef3:383c/64 scope link
       valid_lft forever preferred_lft forever
root@am64xx-evm:~# ip link add link br0 name br0.100 type vlan id 100 egress-qos-map 0:3
.114/24 dev br0.100[  370.479364] 001: 8021q: 802.1Q VLAN Support v1.8
[  370.479435] 001: 8021q: adding VLAN 0 to HW filter on device eth0
root@am64xx-evm:~# [  370.479535] 001: 8021q: adding VLAN 0 to HW filter on device eth1
sleep 1
root@am64xx-evm:~# bridge vlan add dev br0 vid 100 self
root@am64xx-evm:~# bridge vlan add dev eth0 vid 100
root@am64xx-evm:~# bridge vlan add dev eth1 vid 100
root@am64xx-evm:~# ip link set dev br0.100 up
root@am64xx-evm:~# ip addr add 1.1.1.121/24 dev br0.100
root@am64xx-evm:~#
root@am64xx-evm:~# iperf3 -s -p 5201
-----------------------------------------------------------
Server listening on 5201
-----------------------------------------------------------
Accepted connection from 1.1.1.114, port 50644
[  5] local 1.1.1.121 port 5201 connected to 1.1.1.114 port 44179
[ ID] Interval           Transfer     Bitrate         Jitter    Lost/Total Datagrams
[  5]   0.00-1.00   sec  1.19 MBytes  9.99 Mbits/sec  0.004 ms  0/6247 (0%)
[  5]   1.00-2.00   sec  1.19 MBytes  10.0 Mbits/sec  0.009 ms  0/6249 (0%)
[  5]   2.00-3.00   sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6251 (0%)
[  5]   3.00-4.00   sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6250 (0%)
[  5]   4.00-5.00   sec  1.19 MBytes  10.0 Mbits/sec  0.006 ms  0/6250 (0%)
[  5]   5.00-6.00   sec  1.19 MBytes  10.0 Mbits/sec  0.006 ms  0/6249 (0%)
[  5]   6.00-7.00   sec  1.19 MBytes  9.99 Mbits/sec  0.017 ms  0/6246 (0%)
[  5]   7.00-8.00   sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6257 (0%)
[  5]   8.00-9.00   sec  1.19 MBytes  9.99 Mbits/sec  0.008 ms  0/6246 (0%)
[  5]   9.00-10.00  sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6254 (0%)
[  5]  10.00-11.00  sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6247 (0%)
[  5]  11.00-12.00  sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6253 (0%)
[  5]  12.00-13.00  sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6250 (0%)
[  5]  13.00-14.00  sec  1.19 MBytes  10.0 Mbits/sec  0.004 ms  0/6248 (0%)
[  5]  14.00-15.00  sec  1.19 MBytes  10.0 Mbits/sec  0.007 ms  0/6249 (0%)
[  5]  15.00-16.00  sec  1.19 MBytes  10.0 Mbits/sec  0.006 ms  0/6250 (0%)
[  5]  16.00-17.00  sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6251 (0%)
[  5]  17.00-18.00  sec  1.19 MBytes  10.0 Mbits/sec  0.005 ms  0/6250 (0%)
[  5]  18.00-19.00  sec  1.19 MBytes  10.0 Mbits/sec  0.004 ms  0/6250 (0%)
[  5]  19.00-20.00  sec  1.19 MBytes  10.0 Mbits/sec  0.004 ms  0/6249 (0%)
[  5]  20.00-21.00  sec  1.19 MBytes  10.0 Mbits/sec  0.004 ms  0/6253 (0%)

CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | ttyUSB0
```

- See statistics for preemption for the background traffic using:
  - ethtool -S eth0 | grep iet
- And cut-through using:
  - ethtool -S eth1 | grep coll
  - The statistics couners for collisions have been repurposed for cut-through
    - tx_collision_frames: Tx Cut Thru with and without delay (full-duplex)
    - tx_single_coll_frames: Tx Store and Forward (full-duplex)
    - tx_mult_coll_frames: Rx Cut Thru with no delay (full-duplex)
    - tx_excessive_collisions: Rx Cut Thru with delay (full-duplex)
    - tx_late_collisions: Rx Store and Forward (full-duplex)

What about latency and EST (802.1Qbv)

- For EST (802.1Qbv) configuring the schedule across multiple switches requires an engineering tool and configuration stack
  - Building blocks for configuring an EST schedule documented at Sitara SDK EST documentation and Linux TSN documentation on EST
  - Includes endpoint only demo with tcpdump based packet capture and wireshark analysis
- For preemption and cut-through
  - A real realtime application (not iperf3) that for example toggles an IO pin and use an external analyzer
  - Microsecond level accuracy at 1Gbit/s, slowing down the interface to 100M or 10M using ethtool moves the benefit of the features to hundreds microseconds

Summary

- AM64x TSN features
- Demo
  - Configure cut-through switching with preemption
  - IEEE802.1AS-2011 using ptp4l
  - Express and background traffic
- Latency
- Questions and Answers

Resources

- Slides and videos will be posted here:
  - E2E Post
- AM64x GP EVM
- AM64x SK EVM
- AM64x Product Page
- Previous Sitara Webinar on Comms and Real-time
  - Includes Presentation and Video
  - Dives further into the Software Architecture of the Web Server Application
- Previous Sitara Webinar on general Linux Networking
  - Includes Presentation and Video coming soon
  - Covers wireless and wired switching/bridging and support topologies