

Manual:Performance Testing with Traffic Generator

From MikroTik Wiki

Contents

- 1 Summary
- 2 RB1100AHx2 Test setup
 - 2.1 Connecting the routers
 - 2.2 Routing Performance Testing
 - 2.2.1 DUT Config
 - 2.2.2 Traffic Generator Config
 - 2.2.3 Running Tests
 - 2.3 Bridging Performance Testing
 - 2.3.1 DUT Config
 - 2.3.2 Traffic Generator Config
 - 2.3.3 Running Tests
- 3 CCR1036-8G-2S+ Test setup
 - 3.1 Defaults, routing
 - 3.1.1 DUT configuration
 - 3.1.2 TR1 configuration (10Gbit interfaces)
 - 3.1.3 TR2 configuration (1Gbit interfaces)
 - 3.1.4 Running test
- 4 See More

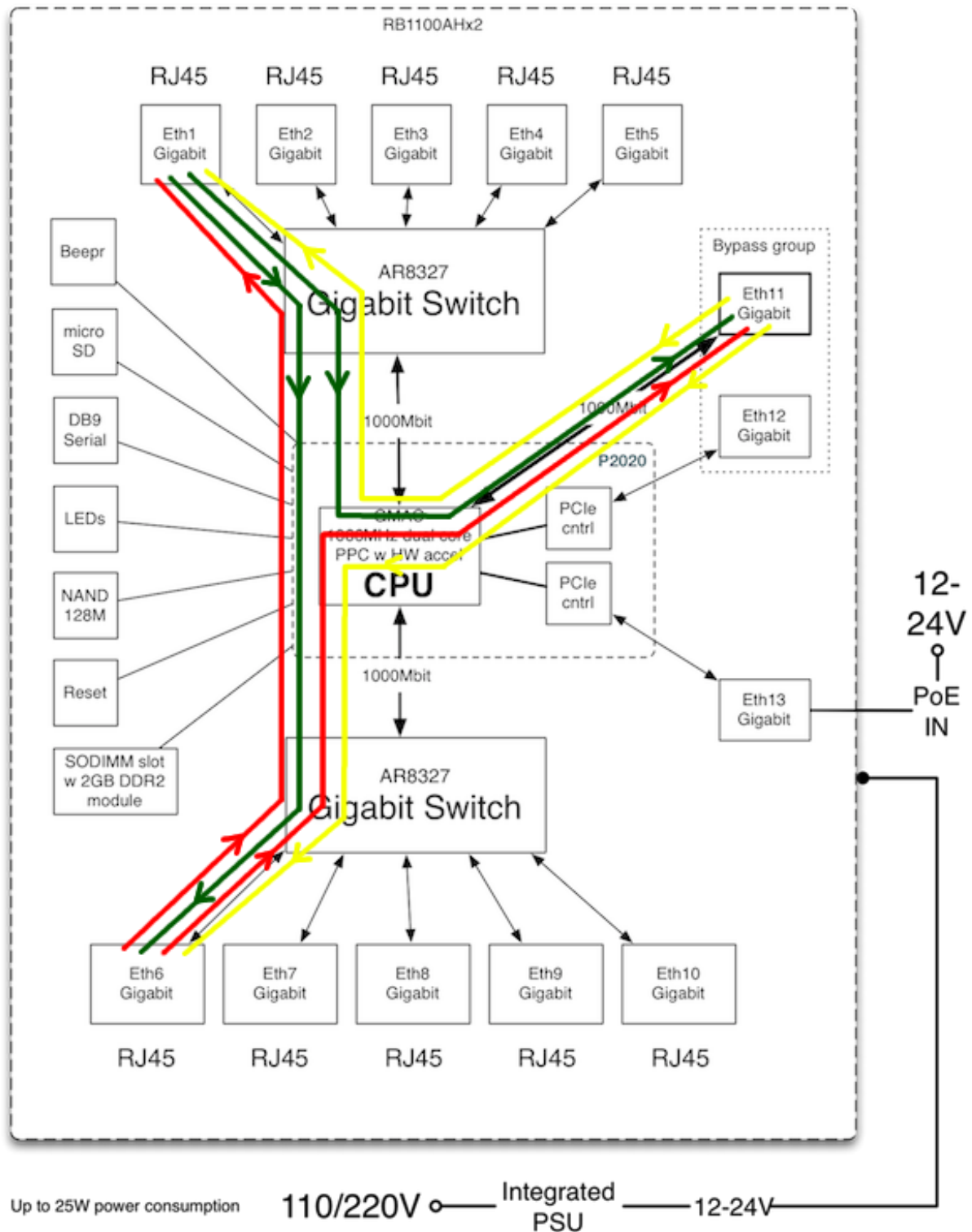
Summary

RouterOS Version 6 introduces a new tool - "traffic generator", which allows to perform performance testing without expensive testing hardware. Traffic is generated from one more router in the network.

This article shows necessary configuration and hardware to replicate the tests published in routerboard.com (<http://routerboard.com/RB1100AHx2#tests>).

RB1100AHx2 Test setup

First step is to choose which ports we will be using for testing.



If we look at the diagram how ports are connected to CPU, fastest combinations are:

- port from switch1 to port form switch chip2,
- ether11 to switch chip,
- ether12/13 to switch chip or to ether11.

To get the maximum out of RB1100AHx2 we will be running 6 streams in total:

- from ether1 to ether6
- from ether1 to ether11
- from ether6 to ether1
- from ether6 to ether11
- from ether11 to ether6
- from ether11 to ether1

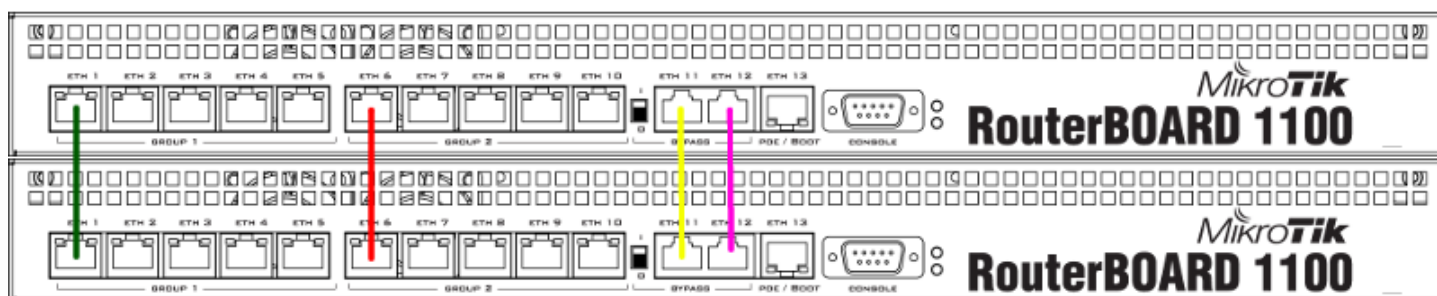
In our test environment one RB1100AHx2 will be device under test (DUT) and other RB1100AHx2 will be a Traffic generator device.

Connecting the routers



Note: RouterOS v6 should be used on both test routers.

Connect cables like this: ether1 to ether1, ether6 to ether6, ether11 to ether11



Note: Ether12 will be added where test reaches wire speed.

Now proceed with software configuration. Either it will be routing (layer3) testing or bridging (layer2) testing.

Routing Performance Testing

DUT Config

```
/ip address
add address=1.1.1.254/24 interface=ether1 network=1.1.1.0
add address=2.2.2.254/24 interface=ether6 network=2.2.2.0
add address=3.3.3.254/24 interface=ether11 network=3.3.3.0
```

Traffic Generator Config

```
/ip address
add address=1.1.1.1/24 interface=ether1 network=1.1.1.0
add address=2.2.2.2/24 interface=ether6 network=2.2.2.0
add address=3.3.3.3/24 interface=ether11 network=3.3.3.0

/tool traffic-generator packet-template
add name=r12 header-stack=mac,ip,udp ip-gateway=1.1.1.254 ip-dst=2.2.2.2
add name=r13 header-stack=mac,ip,udp ip-gateway=1.1.1.254 ip-dst=3.3.3.3
add name=r21 header-stack=mac,ip,udp ip-gateway=2.2.2.254 ip-dst=1.1.1.1
add name=r23 header-stack=mac,ip,udp ip-gateway=2.2.2.254 ip-dst=3.3.3.3
add name=r32 header-stack=mac,ip,udp ip-gateway=3.3.3.254 ip-dst=2.2.2.2
add name=r31 header-stack=mac,ip,udp ip-gateway=3.3.3.254 ip-dst=1.1.1.1
```



Note: To force MAC address re-discovery (on device/configuration change, just apply empty "set" command to necessary packet-templates)

Running Tests

```
/tool traffic-generator
quick tx-template=r12,r13,r21,r23,r31,r32 packet-size=60 mbps=300
```



Note: We are specifying 60 byte packet in traffic generator to get a 64 byte packet on ethernet.

```
[admin@TrafficGen] > /tool traffic-gen quick tx-template=r12,r13,r21,r23,r31,r32 packet-size=60
mbps=120
```

24	0	185	422	91.9Mbps	185	190	88.8Mbps	232	3.0Mbps	16us			
24	1	213	397	105.8Mbps	212	747	102.1Mbps	650	3.7Mbps	10.6u			
24	2	186	245	92.3Mbps	186	185	89.3Mbps	60	3.0Mbps	16.4u			
24	3	213	685	105.9Mbps	212	961	102.2Mbps	724	3.7Mbps	10.8u			
24	4	249	142	119.5Mbps	180	400	86.5Mbps	68	742	32.9Mbps			
24	5	249	141	119.5Mbps	193	158	92.7Mbps	55	983	26.8Mbps			
24	TOT	1	297	032	635.3Mbps	1	170	641	561.9Mbps	126	391	73.4Mbps	10.6u

You can also check in the **DUT** if forwarding is actually happening:

```
[admin@DUT] > /interface monitor-traffic aggregate,ether1,ether6,ether11
name:
rx-packets-per-second: 1 235 620 481 094 487 045 267 469
rx-drops-per-second: 0 0 0 0
rx-errors-per-second: 0 0 0 0
rx-bits-per-second: 593.0Mbps 230.9Mbps 233.7Mbps 128.3Mbps
tx-packets-per-second: 1 233 862 360 750 360 402 512 692
tx-drops-per-second: 0 0 0 0
tx-errors-per-second: 0 0 0 0
tx-bits-per-second: 603.9Mbps 178.9Mbps 178.7Mbps 246.0Mbps
```

After running the test you can see that total throughput of 64byte packets is **1'170'641**pps which is a lot faster than shown in routerboard.com (<http://routerboard.com/RB1100AHx2#tests>) results.

This is because by default fast-path mode is enabled.

Lets enable connection tracking on DUT:

```
/ip firewall connection tracking set enabled=yes
```

And run the test again. As you can see now it is close to advertised pps rate.

46	0	249	793	123.8Mbps	127	410	61.1Mbps	122	383	62.7Mbps	3.22n	
46	1	249	791	123.8Mbps	87	232	41.8Mbps	162	559	82.0Mbps	5.2ms	
46	2	249	792	123.8Mbps	127	424	61.1Mbps	122	368	62.7Mbps	3.15n	
46	3	249	792	123.8Mbps	87	219	41.8Mbps	162	573	82.0Mbps	5.18n	
46	4	249	792	119.9Mbps	40	492	19.4Mbps	209	300	100.4Mbps	5.54n	
46	5	249	791	119.8Mbps	46	736	22.4Mbps	203	055	97.4Mbps	5.41n	
46	TOT	1	498	751	735.3Mbps	516	513	247.9Mbps	982	238	487.4Mbps	3.15n

We can now add more firewall rules, queues and any other configuration and see how much router can actually handle.

Lets add some firewall rules

We will take the customer protection rules from the manual

Start by adding default rules that should present on any firewall:

```
/ip firewall filter
add chain=forward protocol=tcp connection-state=invalid \
    action=drop comment="drop invalid connections"
add chain=forward connection-state=established action=accept \
    comment="allow already established connections"
add chain=forward connection-state=related action=accept \
    comment="allow related connections"
```

We get approximately **18%** less packets

53	TOT	1	492	520	732.3Mbps	435	546	209.0Mbps	1	056	974	523.2Mbps	3.08n
----	-----	---	-----	-----	-----------	-----	-----	-----------	---	-----	-----	-----------	-------

Now add more rules from the manual to see how count of firewall rules affects the performance of the board

```
/ip firewall filter
add chain=forward protocol=icmp action=jump jump-target=icmp

add chain=icmp protocol=icmp icmp-options=0:0 action=accept \
    comment="echo reply"
add chain=icmp protocol=icmp icmp-options=3:0 action=accept \
    comment="net unreachable"
add chain=icmp protocol=icmp icmp-options=3:1 action=accept \
    comment="host unreachable"
add chain=icmp protocol=icmp icmp-options=3:4 action=accept \
    comment="host unreachable fragmentation required"
add chain=icmp protocol=icmp icmp-options=4:0 action=accept \
    comment="allow source quench"
add chain=icmp protocol=icmp icmp-options=8:0 action=accept \
    comment="allow echo request"
add chain=icmp protocol=icmp icmp-options=11:0 action=accept \
    comment="allow time exceed"
add chain=icmp protocol=icmp icmp-options=12:0 action=accept \
    comment="allow parameter bad"
add chain=icmp action=drop comment="deny all other types"
```

33	TOT	1 500 908 736.4Mbps	424 197 203.6Mbps	1 076 711 532.8Mbps 4.07n
----	-----	---------------------	-------------------	---------------------------

There are almost no performance changes. You can add any amount of rules and see that there is only a small influence on performance of the router.

Perform the same test with different packet sizes:

```

/tool traffic-generator
quick tx-template=r12,r13,r21,r23,r31,r32 packet-size=508 mbps=500
/tool traffic-generator
quick tx-template=r12,r13,r21,r23,r31,r32 packet-size=1514 mbps=500

```

If we run the test with 1518 packet size then max throughput will be only **2.9Gbps** This is because wire speed of all interfaces are reached.

We will need to add one more port to our test and add streams.

Connect **ether12** to **ether12** and proceed with configuration

On DUT:

```

/ip address
add address=4.4.4.254/24 interface=ether12

```

On TrafficGen

```

/ip address
add address=4.4.4.4/24 interface=ether12

/tool traffic-generator packet-template
add header-stack=mac,ip,udp ip-dst=4.4.4.4/32 ip-gateway=1.1.1.254 ip-src=1.1.1.1/32 name=\
r14
add header-stack=mac,ip,udp ip-dst=4.4.4.4/32 ip-gateway=2.2.2.254 ip-src=2.2.2.2/32 name=\
r24
add header-stack=mac,ip,udp ip-dst=4.4.4.4/32 ip-gateway=3.3.3.254 ip-src=3.3.3.3/32 name=\
r34
add header-stack=mac,ip,udp ip-dst=1.1.1.1/32 ip-gateway=4.4.4.254 ip-src=4.4.4.4/32 name=\
r41
add header-stack=mac,ip,udp ip-dst=2.2.2.2/32 ip-gateway=4.4.4.254 ip-src=4.4.4.4/32 name=\
r42
add header-stack=mac,ip,udp ip-dst=3.3.3.3/32 ip-gateway=4.4.4.254 ip-src=4.4.4.4/32 name=\
r43

```

And now run the test:

```

/tool traffic-generator quick tx-template=r12,r13,r14,r21,r23,r24,r31,r32,r34,r41,r42,r43 \
packet-size=1514 mbps=350

```

30	6	23 472 284.2Mbps	23 328 282.5Mbps	144 1744.1... 3.22n
30	7	28 890 349.9Mbps	28 741 348.1Mbps	149 1804.6... 1.74n
30	8	28 889 349.9Mbps	26 870 325.4Mbps	2 019 24.4Mbps 984us

30	9	23 455	284.0Mbps	23 083	279.5Mbps	372	4.5Mbps	866us
30	10	28 876	349.7Mbps	28 709	347.7Mbps	167	2.0Mbps	922us
30	11	28 875	349.7Mbps	27 277	330.3Mbps	1 598	19.3Mbps	3.33n
30	TOT	323 389	3.9Gbps	311 743	3.7Gbps	11 646	143.6Mbps	341us

As you can see we get **3.7Gbps**.

And with all firewalls enabled from previous tests we get 2.8Gbps which is approximately **30%** slower:

18	TOT	275 405	3.3Gbps	238 143	2.8Gbps	37 262	453.9Mbps	1.57ms
----	-----	---------	---------	---------	----------------	--------	-----------	--------



Note: mind that speed in quick mode is specified per stream, so if you have two streams per port, you need to send 1/2 of traffic per stream

Bridging Performance Testing

DUT Config

```
/interface bridge add
/interface bridge port
add bridge=bridge1 interface=ether1
add bridge=bridge1 interface=ether6
add bridge=bridge1 interface=ether11
```

Traffic Generator Config

```
/ip address
add address=1.1.1.1/24 interface=ether1 network=1.1.1.0
add address=2.2.2.2/24 interface=ether6 network=2.2.2.0
add address=3.3.3.3/24 interface=ether11 network=3.3.3.0

/tool traffic-generator packet-template
add header-stack=mac,ip,udp ip-src=1.1.1.1/32 ip-dst=2.2.2.2/32 name=b12
add header-stack=mac,ip,udp ip-src=1.1.1.1/32 ip-dst=3.3.3.3/32 name=b13
add header-stack=mac,ip,udp ip-src=2.2.2.2/32 ip-dst=1.1.1.1/32 name=b21
add header-stack=mac,ip,udp ip-src=2.2.2.2/32 ip-dst=3.3.3.3/32 name=b23
add header-stack=mac,ip,udp ip-src=3.3.3.3/32 ip-dst=1.1.1.1/32 name=b31
add header-stack=mac,ip,udp ip-src=3.3.3.3/32 ip-dst=2.2.2.2/32 name=b32
```

Running Tests

```

/tool traffic-generator
quick tx-template=b12,b13,b21,b23,b31,b32 packet-size=60 mbps=200
/tool traffic-generator
quick tx-template=b12,b13,b21,b23,b31,b32 packet-size=508 mbps=500
/tool traffic-generator
quick tx-template=b12,b13,b21,b23,b31,b32 packet-size=1514 mbps=500

```

With small packets we get approximately 1.4 mil packets per second

187	0	195	659	97.0Mbps	195	640	93.9Mbps	19	3.1Mbps	22us
187	1	236	906	117.5Mbps	221	901	106.5Mbps	15 005	10.9Mbps	18.7u
187	2	202	678	100.5Mbps	202	678	97.2Mbps	0	3.2Mbps	18.7u
187	3	238	750	118.4Mbps	231	348	111.0Mbps	7 402	7.3Mbps	12.1u
187	4	263	906	126.6Mbps	256	146	122.9Mbps	7 760	3.7Mbps	23.9u
187	5	263	906	126.6Mbps	256	030	122.8Mbps	7 876	3.7Mbps	14.3u
187	TOT	1 401	805	686.8Mbps	1 363	743	654.5Mbps	38 062	32.2Mbps	12.1u

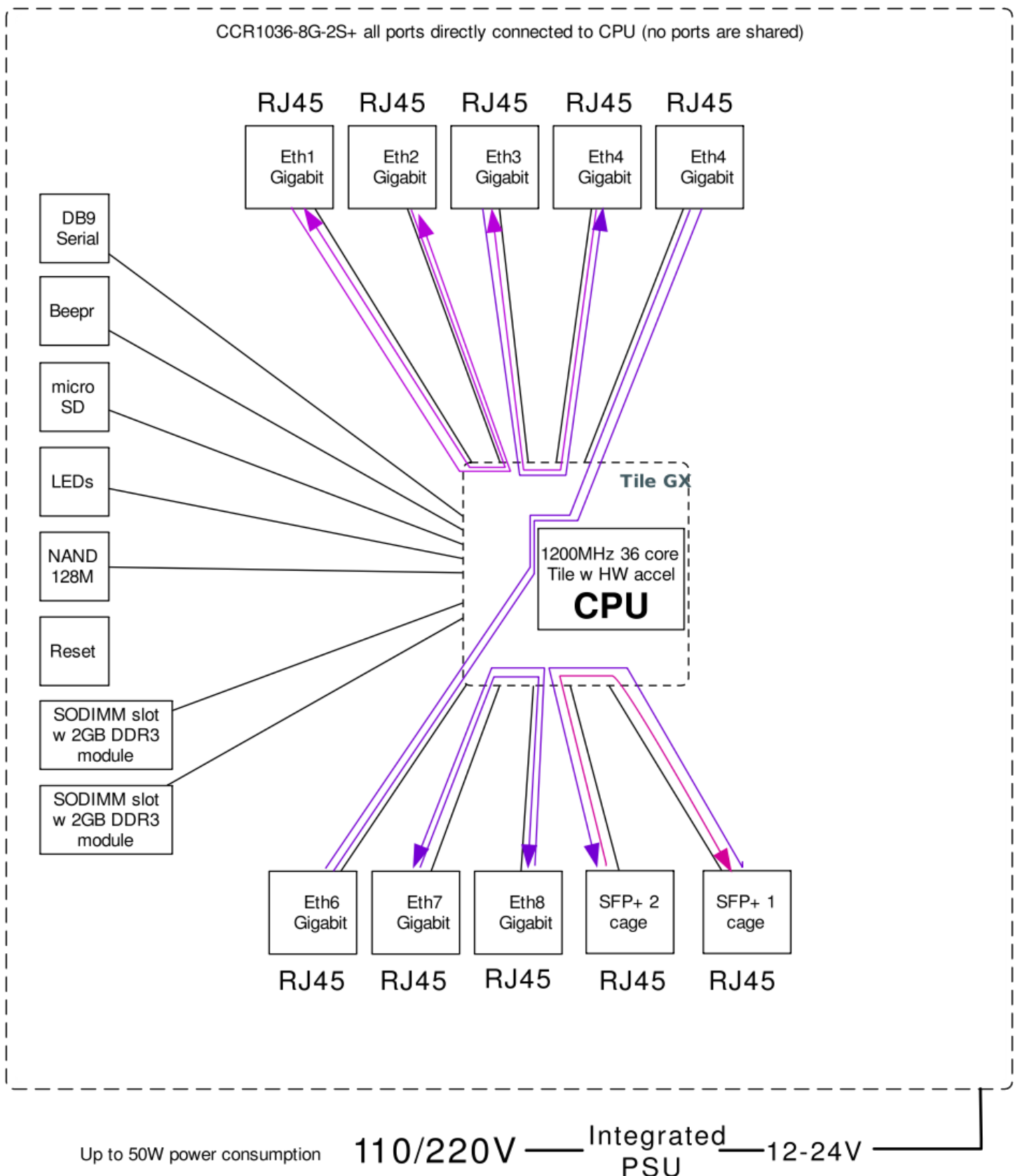
With 1518 byte packets we will get wire speed maximum

11	TOT	243	587	2.9Gbps	241	695	2.9Gbps	1	892	25.5Mbps	1.04n
----	-----	-----	-----	---------	-----	-----	---------	---	-----	----------	-------

So we will need to use ether12 and add few more streams just like in routing test.

CCR1036-8G-2S+ Test setup

CCR1036 series routers have very powerful CPU, So the test will use all ports. As showed in the diagram below



Test will be separated in units, each unit will hold a pair of interfaces. Here is the list of units and what medium is used:

- sfp-sfpplus1 will pass data to sfp-sfpplus2 (direct attach copper 10Gbit)
- ether1 to ether2 (cat5e)

- ether3 to ether4 (cat5e)
- ether5 to ether6 (cat5e)
- ether7 to ether8 (cat5e)

For testing we will require 2 other CCR1036 as passing data through is not as intensive as generating traffic and gathering statistical data.



Note: For test with large packets only one additional CCR1036 is required for use with traffic-generator tool. All the configuration examples will not interfere with each other and can be imported on one testing router

Connect TR1 (Traffic-generator Router 1) to DUT:

- SFP/SFP+1 to SFP/SFP+1 Direct attach 10Gbit cable (10Gbit optical SFP interfaces cable used instead)
- SFP/SFP+2 to SFP/SFP+2 Direct attach 10Gbit cable (10Gbit optical SFP interfaces cable used instead)

Connect TR2 to DUT:

- ether1 to ether1
- ether2 to ether2
- ether3 to ether3
- ether4 to ether4
- ether5 to ether5
- ether6 to ether6
- ether7 to ether7
- ether8 to ether8

Defaults, routing

Default configuration of DUT is with enabled fast-path

DUT configuration

```
/ip address
add address=10.0.100.1/24 interface=sfp-sfpplus1 network=10.0.100.0
add address=10.0.101.1/24 interface=sfp-sfpplus2 network=10.0.101.0
add address=10.0.110.1/24 interface=ether1 network=10.0.110.0
add address=10.0.111.1/24 interface=ether2 network=10.0.111.0
add address=10.0.112.1/24 interface=ether3 network=10.0.112.0
add address=10.0.113.1/24 interface=ether4 network=10.0.113.0
add address=10.0.114.1/24 interface=ether5 network=10.0.114.0
add address=10.0.115.1/24 interface=ether6 network=10.0.115.0
add address=10.0.116.1/24 interface=ether7 network=10.0.116.0
add address=10.0.117.1/24 interface=ether8 network=10.0.117.0
```

TR1 configuration (10Gbit interfaces)

```
/ip address
add address=10.0.100.2/24 interface=sfp-sfpplus1 network=10.0.100.0
add address=10.0.101.2/24 interface=sfp-sfpplus2 network=10.0.101.0
```

```
/tool traffic-generator packet-template
add header-stack=mac,ip,udp interface=sfp-sfpplus1 ip-dst=10.0.101.2 ip-gateway=10.0.100.1 name=pt10
add header-stack=mac,ip,udp interface=sfp-sfpplus2 ip-dst=10.0.100.2 ip-gateway=10.0.101.1 name=pt11
```

```
/tool traffic-generator stream
add id=0 mbps=6000 name=str10 packet-size=60 tx-template=pt10
add id=1 mbps=6000 name=str11 packet-size=60 tx-template=pt11
```

TR2 configuration (1Gbit interfaces)

```
/ip address
add address=10.0.110.2/24 interface=ether1 network=10.0.110.0
add address=10.0.111.2/24 interface=ether2 network=10.0.111.0
add address=10.0.112.2/24 interface=ether3 network=10.0.112.0
add address=10.0.113.2/24 interface=ether4 network=10.0.113.0
add address=10.0.114.2/24 interface=ether5 network=10.0.114.0
add address=10.0.115.2/24 interface=ether6 network=10.0.115.0
add address=10.0.116.2/24 interface=ether7 network=10.0.116.0
add address=10.0.117.2/24 interface=ether8 network=10.0.117.0
```

```
/tool traffic-generator packet-template
add header-stack=mac,ip,udp ip-dst=10.0.111.2 ip-gateway=10.0.110.1 name=pt1
add header-stack=mac,ip,udp ip-dst=10.0.110.2 ip-gateway=10.0.111.1 name=pt2
add header-stack=mac,ip,udp ip-dst=10.0.113.2 ip-gateway=10.0.112.1 name=pt3
add header-stack=mac,ip,udp ip-dst=10.0.112.2 ip-gateway=10.0.113.1 name=pt4
add header-stack=mac,ip,udp ip-dst=10.0.115.2 ip-gateway=10.0.114.1 name=pt5
add header-stack=mac,ip,udp ip-dst=10.0.114.2 ip-gateway=10.0.115.1 name=pt6
add header-stack=mac,ip,udp ip-dst=10.0.117.2 ip-gateway=10.0.116.1 name=pt7
add header-stack=mac,ip,udp ip-dst=10.0.116.2 ip-gateway=10.0.117.1 name=pt8
```

```
/tool traffic-generator stream
add id=0 mbps=700 name=str0 packet-size=60 tx-template=pt0
add id=1 mbps=700 name=str1 packet-size=60 tx-template=pt1
add id=2 mbps=700 name=str3 packet-size=60 tx-template=pt2
add id=3 mbps=700 name=str4 packet-size=60 tx-template=pt3
add id=4 mbps=700 name=str5 packet-size=60 tx-template=pt4
add id=5 mbps=700 name=str6 packet-size=60 tx-template=pt5
add id=6 mbps=700 name=str7 packet-size=60 tx-template=pt6
add id=7 mbps=700 name=str8 packet-size=60 tx-template=pt7
```

Running test

To run tests use ON TR1

```
/tool traffic-generator quick packet-size=60 mbps=6000
```

On TR2

```
/tool traffic-generator quick packet-size=60 mbps=700
```

Profile for medium sized packets:

ON TR1

```
/tool traffic-generator quick packet-size=508 mbps=10000
```

On TR2

```
/tool traffic-generator quick packet-size=508 mbps=1000
```

Profile for large packets: ON TR1

```
/tool traffic-generator quick packet-size=1514 mbps=10000
```

On TR2

```
/tool traffic-generator quick packet-size=1514 mbps=1000
```

See More

- Traffic Generator Manual
- Fast Path

Retrieved from "https://wiki.mikrotik.com/index.php?title=Manual:Performance_Testing_with_Traffic_Generator&oldid=26224"

Categories: [Manual](#) | [Tools](#) | [Examples](#) | [Hardware](#)

- This page was last edited on 27 March 2014, at 16:54.