100BASE-T1-TX-N TEST Report

Menu

1. Test Overview:

Standard network testing typically requires specialized equipment that supports RFC 2544. RFC 2544 is a standard published by the IETF (Internet Engineering Task Force), primarily describing the testing methodology for evaluating the performance of network devices. Its full name is "Benchmarking Methodology for Network Interconnect Devices." This document defines a set of benchmarking tests specifically designed to assess the performance of network devices such as routers and switches.

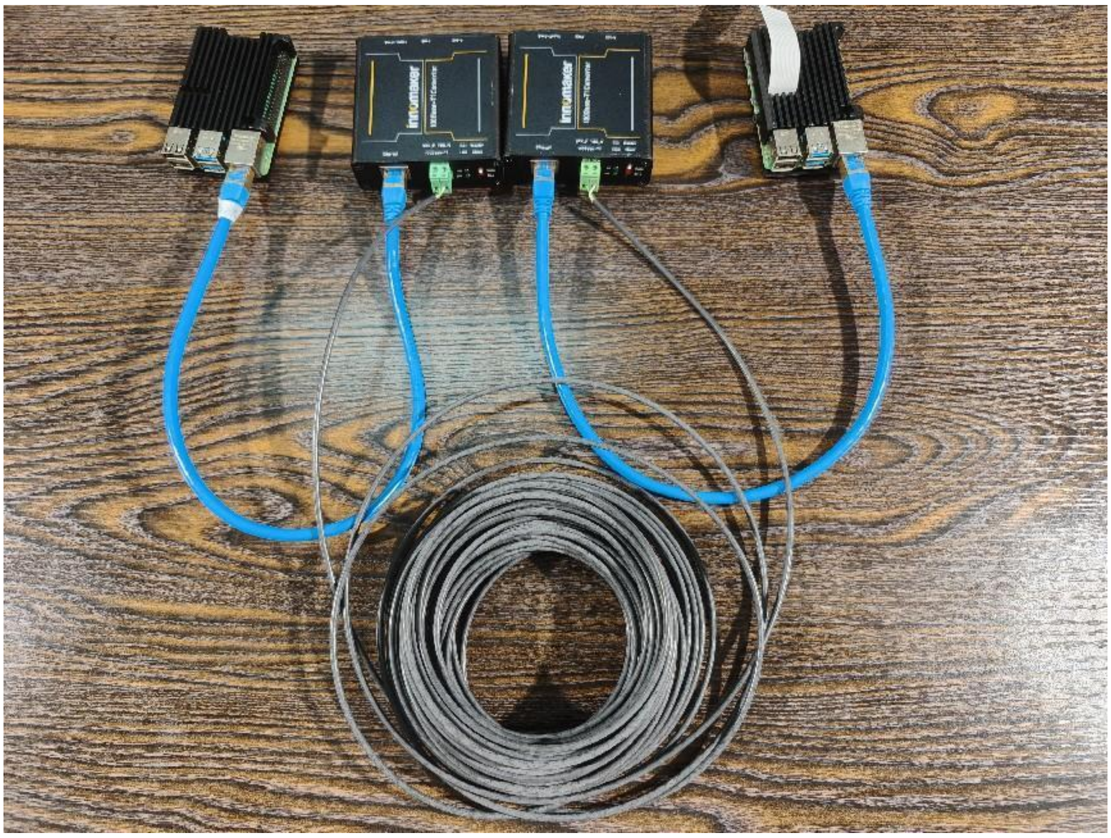
Considering that not all users are willing to purchase these expensive specialized devices, and that they are difficult to set up on-site for simulation analysis, we provide a simpler testing method here. This approach aims to evaluate whether the device's performance and stability can meet your requirements in your specific usage environment.

2. Preparation Before Testing

Here, I use two Raspberry Pi 4 boards as hosts for testing. I prefer using Raspberry Pi for testing because it is affordable and easy to set up. It can be widely used in factories for functional and aging tests. If you don't have a Raspberry Pi 4, other ARM boards or computers with a 100Mbps Ethernet port will work just as well.

2.1 Connect

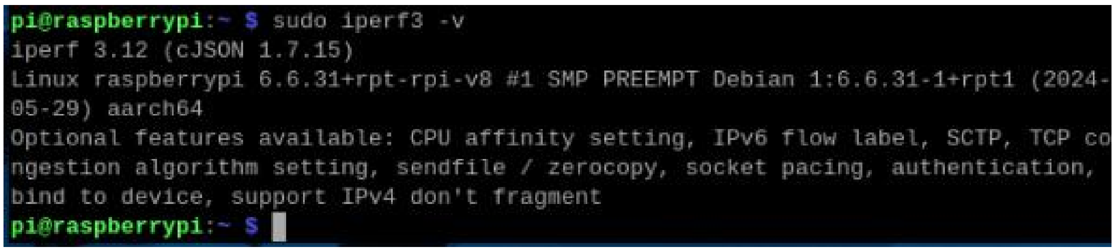
Connect the TX interface of the 100BASE-T1-TX-N module to the Ethernet port of the Raspberry Pi using a network cable, and connect the T1 interfaces of two 100BASE-T1-TX-N modules with a twisted pair unshielded cable. In this case, I am using the German LEONI Dacar647 647-4 automotive 5G gigabit Ethernet twisted pair cable. For the cable specifications, please refer to Appendix 1. Typically, T1 test cables are only used up to 15 meters, but here I am using a 20-meter length to simulate a more challenging test environment.



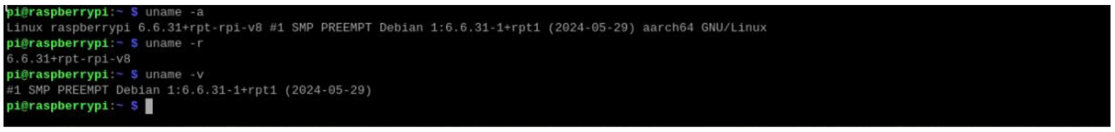
2.2 Software

On the software side, I use the IPERF3 tool, a well-established utility. You can easily find plenty of information about it on Google. Here, I'll only provide the version used and the relevant commands.

I am using Iperf3 version 3.12 on a Raspberry Pi system with version 6.6.31 64-bit. I previously experienced high packet loss and retransmission frequency when using Iperf3 version 3.9 on an older 32-bit Linux system. So, if you encounter similar issues, consider testing with a newer system and a more recent version of Iperf3.



```
pi@raspberrypi:~ $ sudo iperf3 -v
iperf 3.12 (cJSON 1.7.15)
Linux raspberrypi 6.6.31+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.6.31-1+rpt1 (2024-
05-29) aarch64
Optional features available: CPU affinity setting, IPv6 flow label, SCTP, TCP co
ngestion algorithm setting, sendfile / zerocopy, socket pacing, authentication,
bind to device, support IPv4 don't fragment
pi@raspberrypi:~ $
```



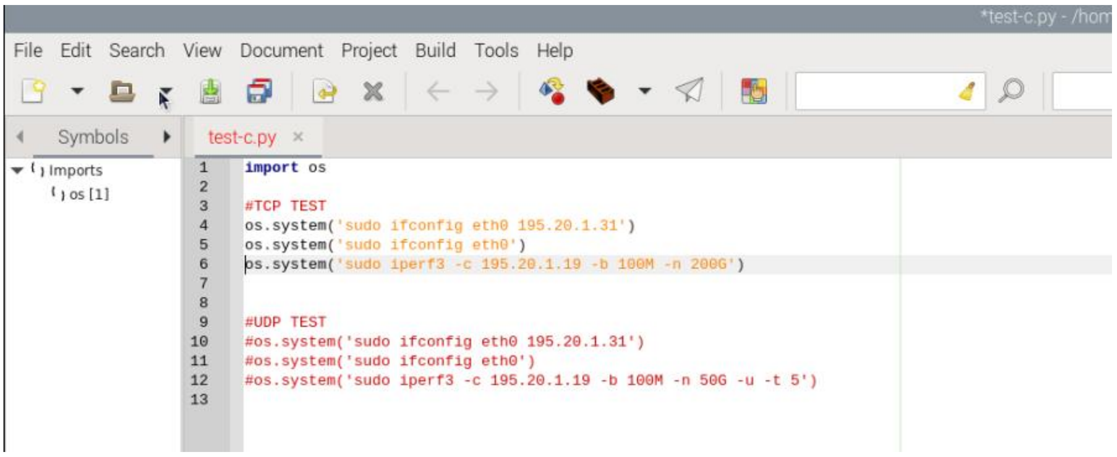
```
pi@raspberrypi:~ $ uname -a
Linux raspberrypi 6.6.31+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.6.31-1+rpt1 (2024-05-29) aarch64 GNU/Linux
pi@raspberrypi:~ $ uname -r
6.6.31+rpt-rpi-v8
pi@raspberrypi:~ $ uname -v
#1 SMP PREEMPT Debian 1:6.6.31-1+rpt1 (2024-05-29)
pi@raspberrypi:~ $
```

3. TCP Testing

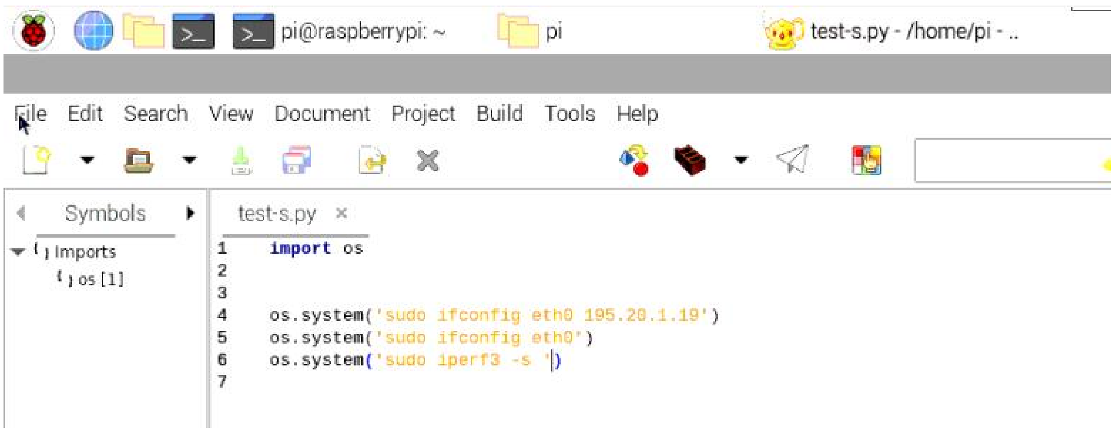
3.1 Starting the test

Start the Raspberry Pi, and in the terminal, set the IP address of the Raspberry Pi using the following commands. Set one Raspberry Pi to run in Server mode and the other one to run in Client mode. Make sure both IP addresses are set in the same subnet but with different IP addresses. Here, I set one to 195.20.1.19 and the other to 195.20.1.31.

To make repeated testing easier, I've written these commands into Python scripts named test-c.py and test-s.py.



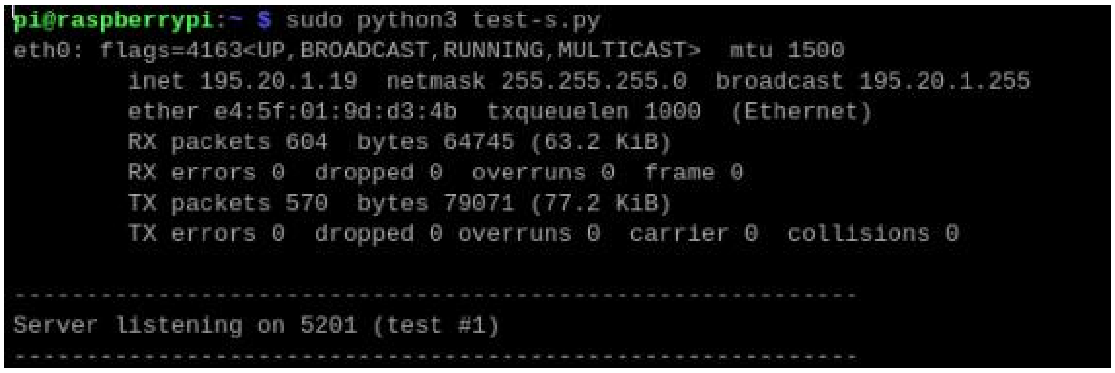
```
import os

#TCP TEST
os.system('sudo ifconfig eth0 195.20.1.31')
os.system('sudo ifconfig eth0')
os.system('sudo iperf3 -c 195.20.1.19 -b 100M -n 200G')


#UDP TEST
#os.system('sudo ifconfig eth0 195.20.1.31')
#os.system('sudo ifconfig eth0')
#os.system('sudo iperf3 -c 195.20.1.19 -b 100M -n 50G -u -t 5')

```


pi@raspberrypi: ~
pi
test-s.py - /home/pi - ..
File Edit Search View Document Project Build Tools Help
Symbols
Imports
os [1]
test-s.py

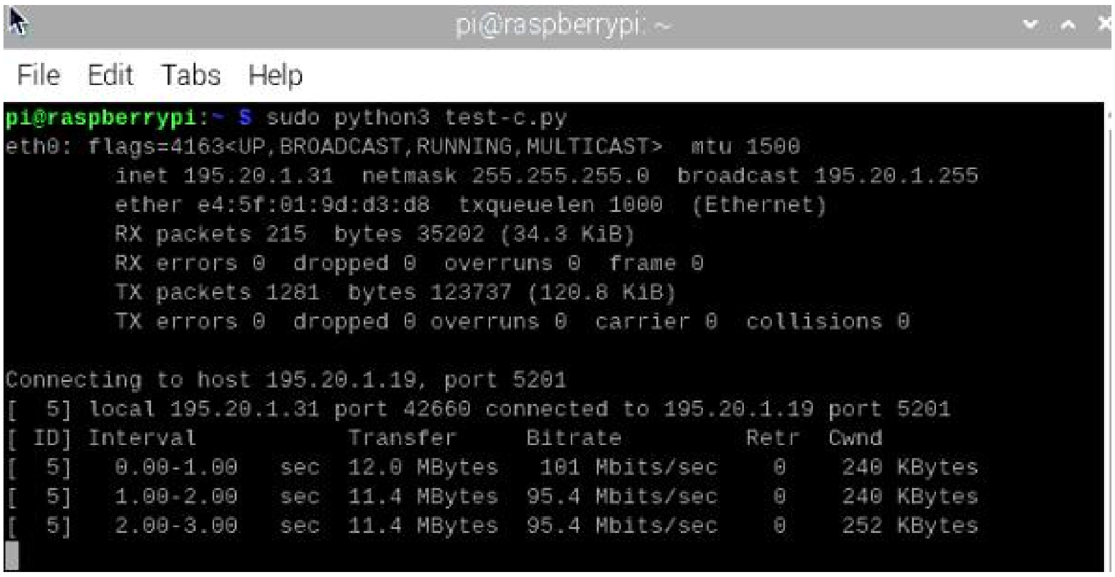
```
import os


os.system('sudo ifconfig eth0 195.20.1.19')
os.system('sudo ifconfig eth0')
os.system('sudo iperf3 -s ')

```

Then, in sequence, first run sudo python3 test-s.py and then run sudo python3 test-c.py. After the execution, you will see data being printed, followed by waiting for the test results. The -n 200G parameter represents continuously testing 200GB of data. You can adjust this data amount according to your own needs.

Note that during this test, do not operate the Raspberry Pi, especially opening web pages or similar activities, as it may affect the test speed and packet loss rate.



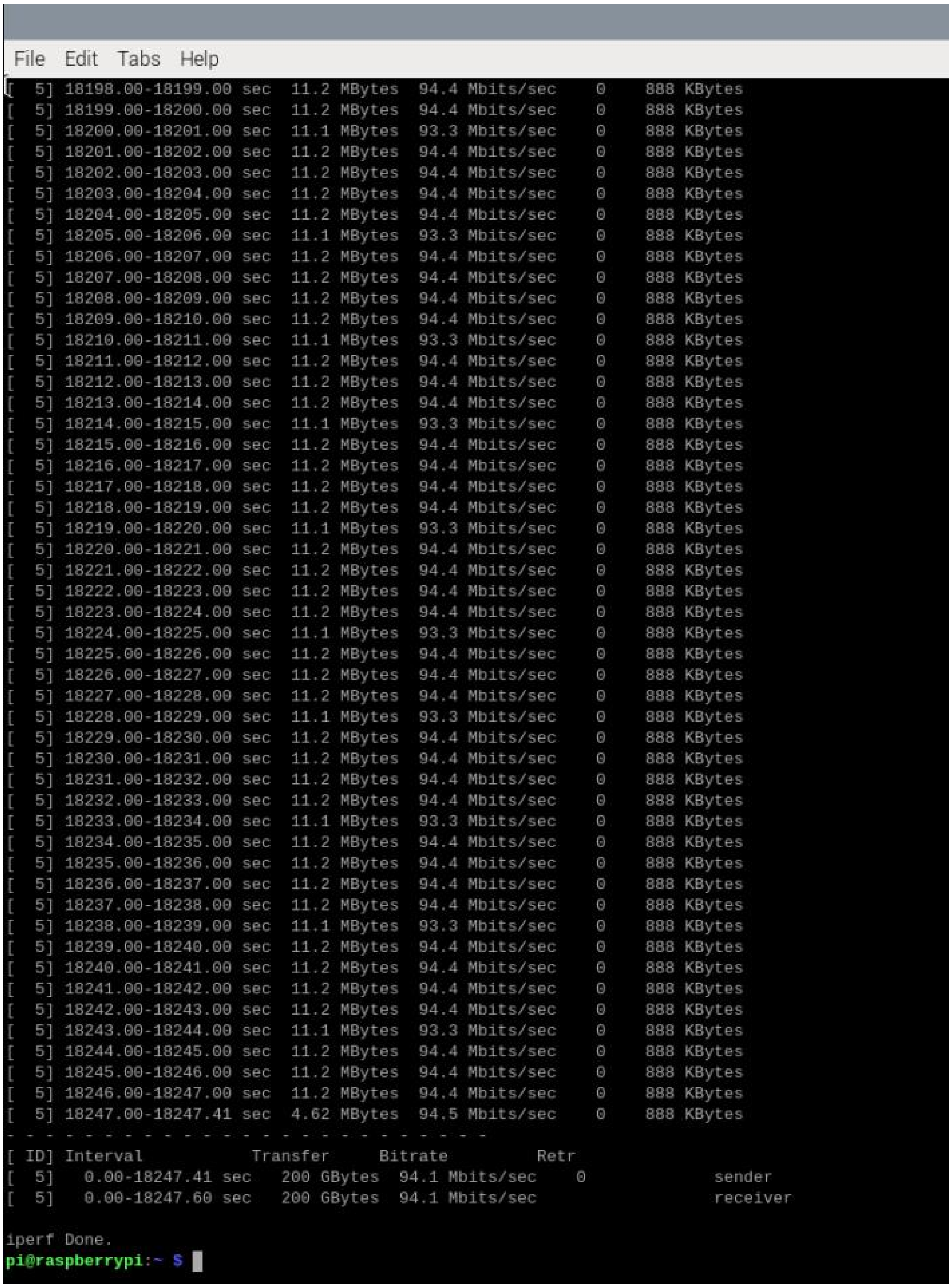
```
pi@raspberrypi:~ $ sudo python3 test-s.py
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 195.20.1.19  netmask 255.255.255.0  broadcast 195.20.1.255
        ether e4:5f:01:9d:d3:4b  txqueuelen 1000  (Ethernet)
        RX packets 604  bytes 64745 (63.2 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 570  bytes 79071 (77.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

-----------------------------------------------------------
Server listening on 5201 (test #1)
-----------------------------------------------------------
```



```
pi@raspberrypi:~ $ sudo python3 test-c.py
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 195.20.1.31  netmask 255.255.255.0  broadcast 195.20.1.255
        ether e4:5f:01:9d:d3:d8  txqueuelen 1000  (Ethernet)
        RX packets 215  bytes 35202 (34.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1281  bytes 123737 (120.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

Connecting to host 195.20.1.19, port 5201
[  5] local 195.20.1.31 port 42660 connected to 195.20.1.19 port 5201
[ ID] Interval           Transfer     Bitrate         Retr  Cwnd
[  5]   0.00-1.00   sec  12.0 MBytes   101 Mbits/sec    0    240 KBytes
[  5]   1.00-2.00   sec  11.4 MBytes  95.4 Mbits/sec    0    240 KBytes
[  5]   2.00-3.00   sec  11.4 MBytes  95.4 Mbits/sec    0    252 KBytes
```

3.2 Test Result

By checking the print message, we can see an average of 94.1 Mbit/sec with 0 retransmissions.



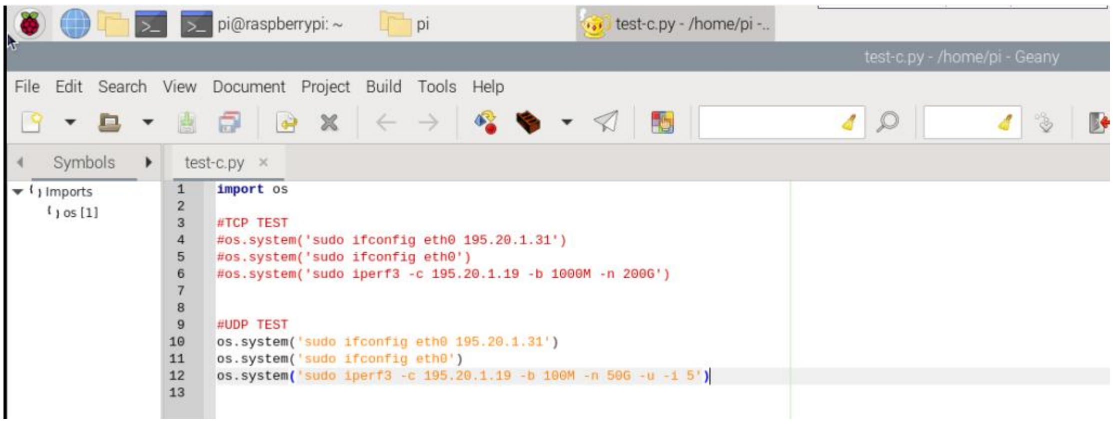
```
[  5] 18198.00-18199.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18199.00-18200.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18200.00-18201.00 sec  11.1 MBytes  93.3 Mbits/sec    0    888 KBytes
[  5] 18201.00-18202.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18202.00-18203.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18203.00-18204.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18204.00-18205.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18205.00-18206.00 sec  11.1 MBytes  93.3 Mbits/sec    0    888 KBytes
[  5] 18206.00-18207.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18207.00-18208.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18208.00-18209.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18209.00-18210.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18210.00-18211.00 sec  11.1 MBytes  93.3 Mbits/sec    0    888 KBytes
[  5] 18211.00-18212.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18212.00-18213.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18213.00-18214.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18214.00-18215.00 sec  11.1 MBytes  93.3 Mbits/sec    0    888 KBytes
[  5] 18215.00-18216.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18216.00-18217.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18217.00-18218.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18218.00-18219.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18219.00-18220.00 sec  11.1 MBytes  93.3 Mbits/sec    0    888 KBytes
[  5] 18220.00-18221.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18221.00-18222.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18222.00-18223.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18223.00-18224.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18224.00-18225.00 sec  11.1 MBytes  93.3 Mbits/sec    0    888 KBytes
[  5] 18225.00-18226.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18226.00-18227.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18227.00-18228.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18228.00-18229.00 sec  11.1 MBytes  93.3 Mbits/sec    0    888 KBytes
[  5] 18229.00-18230.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18230.00-18231.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18231.00-18232.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18232.00-18233.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18233.00-18234.00 sec  11.1 MBytes  93.3 Mbits/sec    0    888 KBytes
[  5] 18234.00-18235.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18235.00-18236.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18236.00-18237.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18237.00-18238.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18238.00-18239.00 sec  11.1 MBytes  93.3 Mbits/sec    0    888 KBytes
[  5] 18239.00-18240.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18240.00-18241.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18241.00-18242.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18242.00-18243.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18243.00-18244.00 sec  11.1 MBytes  93.3 Mbits/sec    0    888 KBytes
[  5] 18244.00-18245.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18245.00-18246.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18246.00-18247.00 sec  11.2 MBytes  94.4 Mbits/sec    0    888 KBytes
[  5] 18247.00-18247.41 sec  4.62 MBytes  94.5 Mbits/sec    0    888 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Retr
[  5]   0.00-18247.41 sec   200 GBytes  94.1 Mbits/sec    0             sender
[  5]   0.00-18247.60 sec   200 GBytes  94.1 Mbits/sec                  receiver

iperf Done.
pi@raspberrypi:~ $ 
```

4. UDP Testing

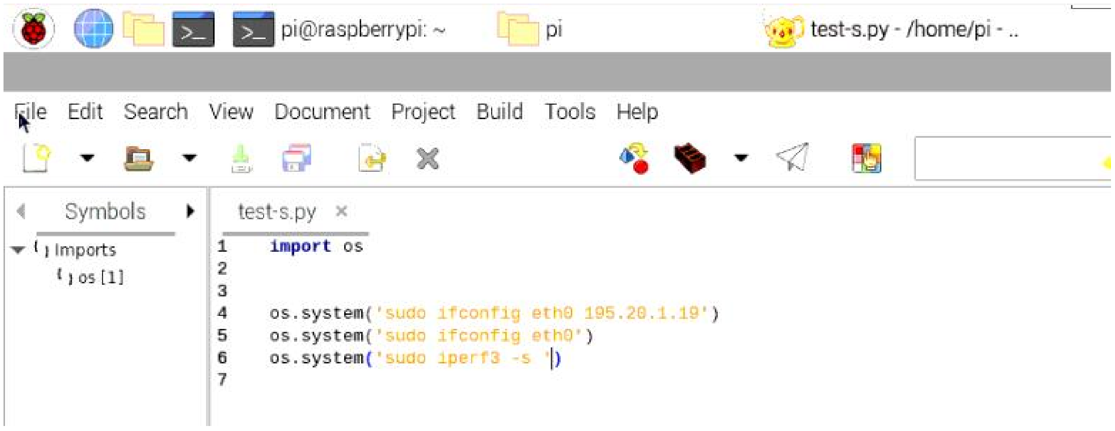
4.1 Starting the test

Start the Raspberry Pi, and in the terminal, set the IP address of the Raspberry Pi using the following commands. Set one Raspberry Pi to run in Server mode and the other one to run in Client mode. Make sure both IP addresses are set in the same subnet but with different IP addresses. Here, I set one to 195.20.1.19 and the other to 195.20.1.31.

To make repeated testing easier, I've written these commands into Python scripts named test-c.py and test-s.py.


pi@raspberrypi: ~
pi
test-c.py - /home/pi -..
test-c.py - /home/pi - Geany
File Edit Search View Document Project Build Tools Help
Symbols
test-c.py
Imports
os [1]

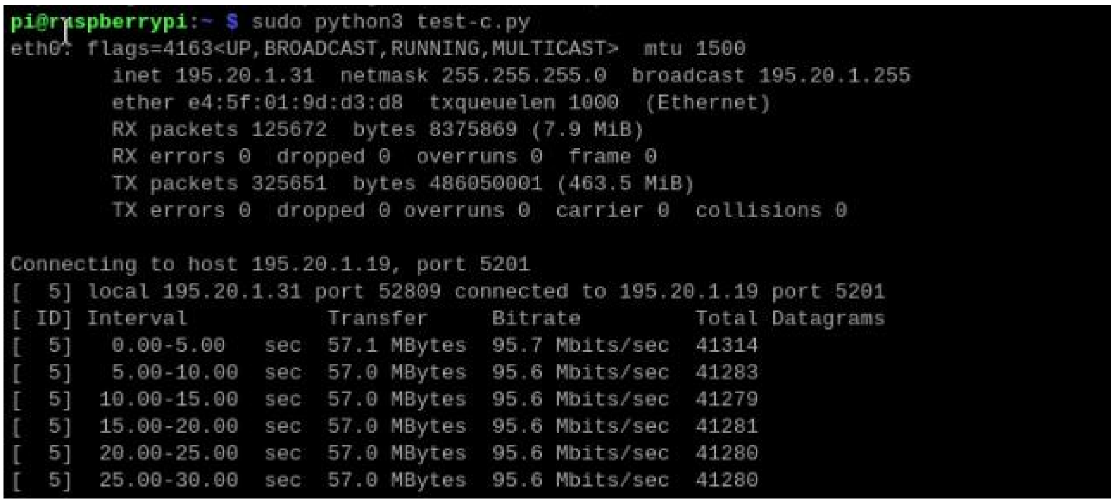
```
import os

#TCP TEST
#os.system('sudo ifconfig eth0 195.20.1.31')
#os.system('sudo ifconfig eth0')
#os.system('sudo iperf3 -c 195.20.1.19 -b 1000M -n 200G')


#UDP TEST
os.system('sudo ifconfig eth0 195.20.1.31')
os.system('sudo ifconfig eth0')
os.system('sudo iperf3 -c 195.20.1.19 -b 100M -n 50G -u -i 5')
```


pi@raspberrypi: ~
pi
test-s.py - /home/pi - ..
File Edit Search View Document Project Build Tools Help
Symbols
test-s.py
Imports
os [1]

```
import os


os.system('sudo ifconfig eth0 195.20.1.19')
os.system('sudo ifconfig eth0')
os.system('sudo iperf3 -s ')
```

Then, in sequence, first run sudo python3 test-s.py and then run sudo python3 test-c.py. After the execution, you will see data being printed, followed by waiting for the test results. The -n 50G parameter represents continuously testing 50GB of data. You can adjust this data amount according to your own needs.

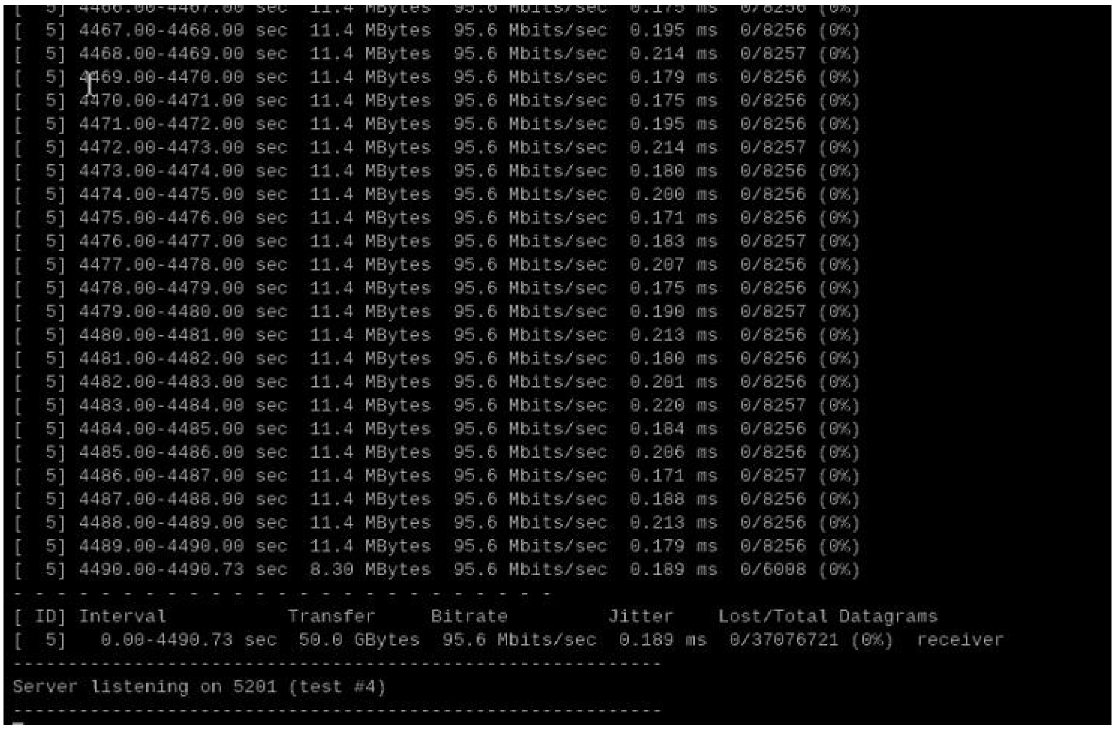
Note that during this test, do not operate the Raspberry Pi, especially opening web pages or similar activities, as it may affect the test speed and packet loss rate.



```
pi@raspberrypi:~ $ sudo python3 test-c.py
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 195.20.1.31  netmask 255.255.255.0  broadcast 195.20.1.255
        ether e4:5f:01:9d:d3:d8  txqueuelen 1000  (Ethernet)
        RX packets 125672  bytes 8375869 (7.9 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 325651  bytes 486050001 (463.5 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

Connecting to host 195.20.1.19, port 5201
[  5] local 195.20.1.31 port 52809 connected to 195.20.1.19 port 5201
[ ID] Interval           Transfer     Bitrate         Total Datagrams
[  5]   0.00-5.00   sec  57.1 MBytes  95.7 Mbits/sec  41314
[  5]   5.00-10.00  sec  57.0 MBytes  95.6 Mbits/sec  41283
[  5]  10.00-15.00  sec  57.0 MBytes  95.6 Mbits/sec  41279
[  5]  15.00-20.00  sec  57.0 MBytes  95.6 Mbits/sec  41281
[  5]  20.00-25.00  sec  57.0 MBytes  95.6 Mbits/sec  41280
[  5]  25.00-30.00  sec  57.0 MBytes  95.6 Mbits/sec  41280
```

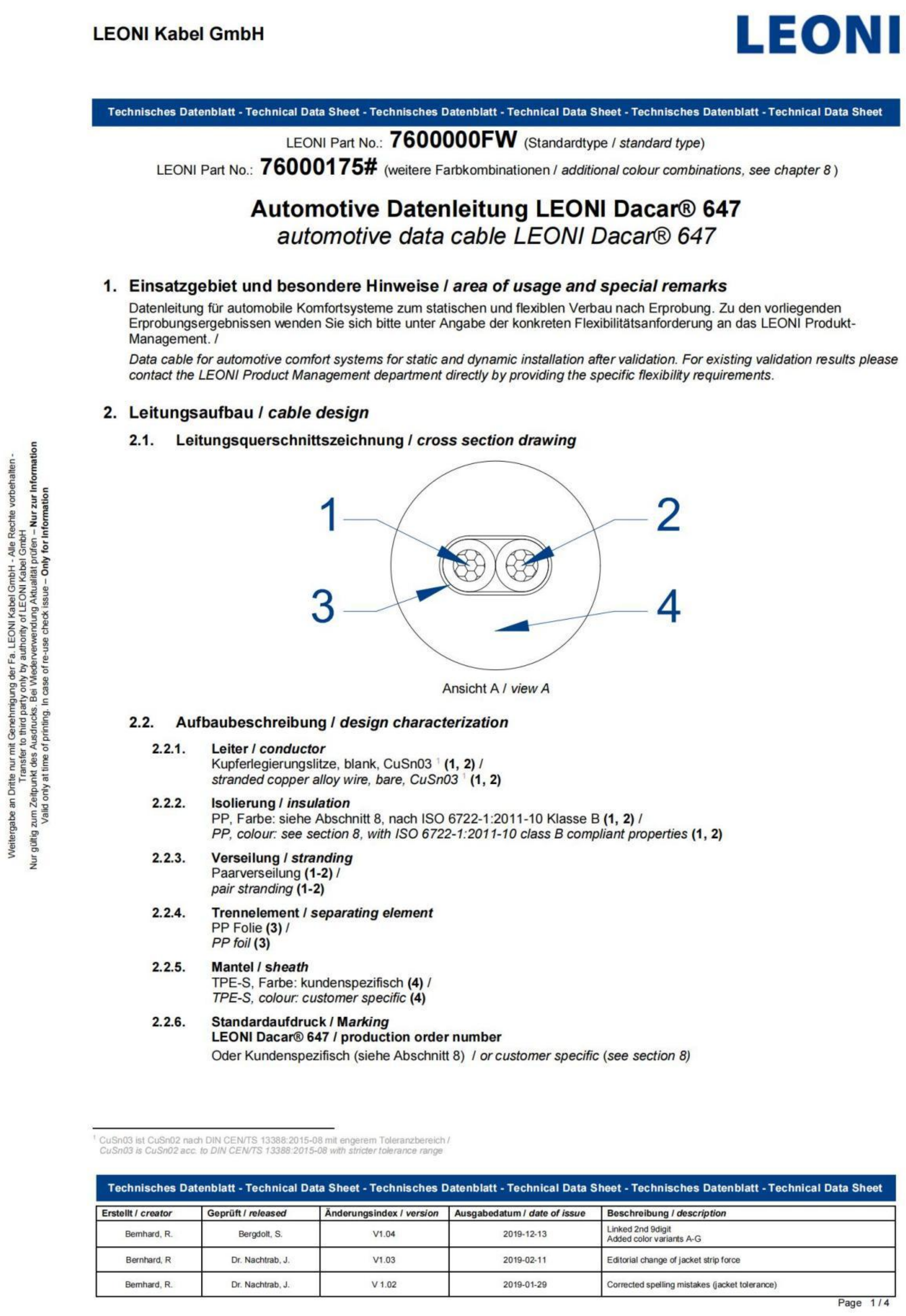
4.2 Test Result

By checking the print message, we can see an average of 95.6 Mbit/sec, 0 lost ,0.189ms jetter.



```
[  5] 4466.00-4467.00 sec  11.4 MBytes  95.6 Mbits/sec  0.175 ms  0/8256 (0%)
[  5] 4467.00-4468.00 sec  11.4 MBytes  95.6 Mbits/sec  0.195 ms  0/8256 (0%)
[  5] 4468.00-4469.00 sec  11.4 MBytes  95.6 Mbits/sec  0.214 ms  0/8257 (0%)
[  5] 4469.00-4470.00 sec  11.4 MBytes  95.6 Mbits/sec  0.179 ms  0/8256 (0%)
[  5] 4470.00-4471.00 sec  11.4 MBytes  95.6 Mbits/sec  0.175 ms  0/8256 (0%)
[  5] 4471.00-4472.00 sec  11.4 MBytes  95.6 Mbits/sec  0.195 ms  0/8256 (0%)
[  5] 4472.00-4473.00 sec  11.4 MBytes  95.6 Mbits/sec  0.214 ms  0/8257 (0%)
[  5] 4473.00-4474.00 sec  11.4 MBytes  95.6 Mbits/sec  0.180 ms  0/8256 (0%)
[  5] 4474.00-4475.00 sec  11.4 MBytes  95.6 Mbits/sec  0.200 ms  0/8256 (0%)
[  5] 4475.00-4476.00 sec  11.4 MBytes  95.6 Mbits/sec  0.171 ms  0/8256 (0%)
[  5] 4476.00-4477.00 sec  11.4 MBytes  95.6 Mbits/sec  0.183 ms  0/8257 (0%)
[  5] 4477.00-4478.00 sec  11.4 MBytes  95.6 Mbits/sec  0.207 ms  0/8256 (0%)
[  5] 4478.00-4479.00 sec  11.4 MBytes  95.6 Mbits/sec  0.175 ms  0/8256 (0%)
[  5] 4479.00-4480.00 sec  11.4 MBytes  95.6 Mbits/sec  0.190 ms  0/8257 (0%)
[  5] 4480.00-4481.00 sec  11.4 MBytes  95.6 Mbits/sec  0.213 ms  0/8256 (0%)
[  5] 4481.00-4482.00 sec  11.4 MBytes  95.6 Mbits/sec  0.180 ms  0/8256 (0%)
[  5] 4482.00-4483.00 sec  11.4 MBytes  95.6 Mbits/sec  0.201 ms  0/8256 (0%)
[  5] 4483.00-4484.00 sec  11.4 MBytes  95.6 Mbits/sec  0.220 ms  0/8257 (0%)
[  5] 4484.00-4485.00 sec  11.4 MBytes  95.6 Mbits/sec  0.184 ms  0/8256 (0%)
[  5] 4485.00-4486.00 sec  11.4 MBytes  95.6 Mbits/sec  0.206 ms  0/8256 (0%)
[  5] 4486.00-4487.00 sec  11.4 MBytes  95.6 Mbits/sec  0.171 ms  0/8257 (0%)
[  5] 4487.00-4488.00 sec  11.4 MBytes  95.6 Mbits/sec  0.188 ms  0/8256 (0%)
[  5] 4488.00-4489.00 sec  11.4 MBytes  95.6 Mbits/sec  0.213 ms  0/8256 (0%)
[  5] 4489.00-4490.00 sec  11.4 MBytes  95.6 Mbits/sec  0.179 ms  0/8256 (0%)
[  5] 4490.00-4490.73 sec  8.30 MBytes  95.6 Mbits/sec  0.189 ms  0/6008 (0%)
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Jitter    Lost/Total Datagrams
[  5]   0.00-4490.73 sec  50.0 GBytes  95.6 Mbits/sec  0.189 ms  0/37076721 (0%)  receiver
-----------------------------------------------------------
Server listening on 5201 (test #4)
-----------------------------------------------------------
```

5. Appendix



LEONI Kabel GmbH

LEONI

Technisches Datenblatt - Technical Data Sheet - Technisches Datenblatt - Technical Data Sheet - Technisches Datenblatt - Technical Data Sheet

LEONI Part No.: **7600000FW** (Standardtype / *standard type*)

LEONI Part No.: **76000175#** (weitere Farbkombinationen / *additional colour combinations, see chapter 8*)

Automotive Datenleitung LEONI Dacar® 647

automotive data cable LEONI Dacar® 647

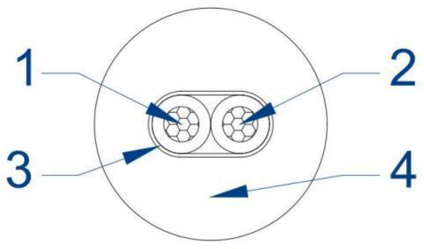
1. Einsatzgebiet und besondere Hinweise / *area of usage and special remarks*

Datenleitung für automobile Komfortsysteme zum statischen und flexiblen Verbau nach Erprobung. Zu den vorliegenden Erprobungsergebnissen wenden Sie sich bitte unter Angabe der konkreten Flexibilitätsanforderung an das LEONI Produkt-Management. /

Data cable for automotive comfort systems for static and dynamic installation after validation. For existing validation results please contact the LEONI Product Management department directly by providing the specific flexibility requirements.

2. Leitungsaufbau / *cable design*

2.1. Leitungsquerschnittszeichnung / *cross section drawing*



Ansicht A / *view A*

2.2. Aufbaubeschreibung / *design characterization*

2.2.1. Leiter / *conductor*
Kupferlegierungslitze, blank, CuSn03 [1] **(1, 2)** /
stranded copper alloy wire, bare, CuSn03 [1] **(1, 2)**

2.2.2. Isolierung / *insulation*
PP, Farbe: siehe Abschnitt 8, nach ISO 6722-1:2011-10 Klasse B **(1, 2)** /
PP, colour: see section 8, with ISO 6722-1:2011-10 class B compliant properties **(1, 2)**

2.2.3. Verseilung / *stranding*
Paarverseilung **(1-2)** /
pair stranding **(1-2)**

2.2.4. Trennelement / *separating element*
PP Folie **(3)** /
PP foil **(3)**

2.2.5. Mantel / *sheath*
TPE-S, Farbe: kundenspezifisch **(4)** /
TPE-S, colour: customer specific **(4)**

2.2.6. Standardaufdruck / *Marking*
LEONI Dacar® 647 / production order number
Oder Kundenspezifisch (siehe Abschnitt 8) / *or customer specific* (*see section 8)*

[1] CuSn03 ist CuSn02 nach DIN CEN/TS 13388:2015-08 mit engerem Toleranzbereich /
CuSn03 is CuSn02 acc. to DIN CEN/TS 13388:2015-08 with stricter tolerance range

Technisches Datenblatt - Technical Data Sheet - Technisches Datenblatt - Technical Data Sheet - Technisches Datenblatt - Technical Data Sheet

Erstellt / *creator*	Geprüft / *released*	Änderungsindex / *version*	Ausgabedatum / *date of issue*	Beschreibung / *description*
Bernhard, R.	Bergdolt, S.	V1.04	2019-12-13	Linked 2nd 9digit Added color variants A-G
Bernhard, R	Dr. Nachtrab, J.	V1.03	2019-02-11	Editorial change of jacket strip force
Bernhard, R.	Dr. Nachtrab, J.	V 1.02	2019-01-29	Corrected spelling mistakes (jacket tolerance)

Page 1 / 4

6.Version Descriptions

Version	Description	Date	E-mail
V1.0		2024.10.16	support@inno-maker.com sales@inno-maker.com

If you have any suggestions, ideas, codes and tools please feel free to email to me. Look forward to your letter and kindly share.