



STM32 MCUs (/s/group/0F90X00000AXsASAW/) (Archived) — Piranha (/s/profile/0050X0000088kkUQAQ) (Customer) asked a question.

[Edited December 3, 2019 at 10:28 AM \(/s/question/0D50X0000AhNB0WSQW/actually-working-stm32-ethernet-and-lwip-demonstration-firmware\)](#)

Actually working STM32 Ethernet and lwIP demonstration firmware

NOTE! For an instruction on how to make Ethernet and lwIP working on STM32 look here: <https://community.st.com/s/question/0D50X0000B0ffhnSQB>
(<https://community.st.com/s/question/0D50X0000B0ffhnSQB>).

Years go by and ST is still incapable of providing an actually working Ethernet and lwIP example for STM32 microcontrollers, especially on Cortex-M7 based series and with RTOS involved. So I decided to do it. The purpose of this is to demonstrate stability, performance and functional flexibility of what ETH peripheral and lwIP IP stack are capable of with properly implemented firmware. The firmware is provided as a compiled HEX file for several STM32F7 series ready-made boards.

How it's built

Components and their sources:

- CMSIS-Core header files: https://github.com/ARM-software/CMSIS_5 (https://github.com/ARM-software/CMSIS_5).
- CMSIS-Device header files: <https://www.st.com/en/embedded-software/stm32cube7.html> (<https://www.st.com/en/embedded-software/stm32cube7.html>).
- FreeRTOS: <https://www.freertos.org/> (<https://www.freertos.org/>).
- lwIP: <https://savannah.nongnu.org/projects/lwip/> (<https://savannah.nongnu.org/projects/lwip/>).
- Drivers, other components, component integration and application developed by me.

All built with GCC compiler at -O2 optimization level within EmBitz IDE (<http://www.embitz.org/>) (<http://www.embitz.org/>).

Features and usage

- Idle thread puts CPU into sleep mode.
- Device status LED blinking once per second deliberately done with a RTOS software timer to ensure correct system operation. The timer thread runs as a second lowest priority thread next to the idle thread.
- Debug output printing through the on-board ST-LINK virtual serial port at 115200/8-N-1. Pressing any key in a terminal enables/disables printing CPU usage once per second. CPU usage includes interrupt processing and context switch times.
- Link detection and autonegotiation. Physical link can be connected/disconnected at any time to any network. Link up/down status indicated by LED.
- Static almost unique locally administered MAC address is generated from a 96-bit unique device ID.
- DHCP client. Look for the assigned IP address at debug output or router's configuration interface.
- HTTP statistics server. Open http://IP_address/ (http://IP_address/) in a web browser.
- Iperf2 server. Use Iperf2 (<https://sourceforge.net/projects/iperf2/>) (<https://sourceforge.net/projects/iperf2/>) tool with options (<https://iperf.fr/iperf-doc.php#doc>) (<https://iperf.fr/iperf-doc.php#doc>) to test TCP performance and watch CPU usage while testing.

Iperf2 command line examples

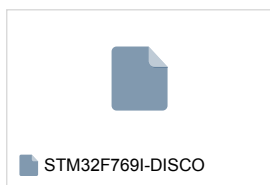
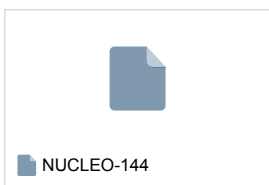
- Rx: `iperf -c IP_address`
- Rx & Tx half-duplex: `iperf -c IP_address -r`
- Rx & Tx full-duplex: `iperf -c IP_address -d`

Measured TCP performance on STM32F76x

- Rx: 94,9 Mbps @ 16,9 % CPU.
- Tx: 94,9 Mbps @ 20,7 % CPU.
- Rx + Tx: 90 + 90 = 180 Mbps @ 33 % CPU.

Boards supported

- NUCLEO-144: NUCLEO-F767ZI and NUCLEO-F746ZG/F756ZG
- STM32F769I-DISCO



[STM32 MCUs](#)

(</s/topic/0TO0X00000BSqSWAW/>)

[FreeRTOS](#)

(</s/topic/0TO0X00000BWTWA4/>)

[Ethernet](#)

(</s/topic/0TO0X00000BWXQWA4/>)

[STM32F7](#)

(</s/topic/0TO0X00000BWTOWA4/>)

[LwIP](#)

(</s/topic/0TO0X00000BTnOWAW/>)

57 answers 29.35K views



Like



Answer



Share

D'avid (/s/profile/0053W000001svo6QAA), GreenGuy (/s/profile/0050X000007vvFOQAY), and [13 others](#) like this.



[kold](/s/profile/0050X00000899t2QAA/) (/s/profile/0050X00000899t2QAA) (Customer)

4 years ago

The performance is quite impressive and stable too!
Is it possible to get insight into how you have done it?

Like Reply




[Piranha](/s/profile/0050X0000088kkUQAQ/) (/s/profile/0050X0000088kkUQAQ) (Customer)

4 years ago

I will not provide source code as a whole, but I'm thinking about making a series of topics about ST's Ethernet and lwIP implementation bugs, limitations and stupid code.

Also I will add STM32F746G-DISCO board soon and later maybe even STM32F7 evaluation boards. By the way it seems that NUCLEO-F767ZI build should work on NUCLEO-F746ZG/F756ZG - someone could test it and report results.


Like Reply 1 like

 [rtek1000 \(/s/profile/0050X000008AC1uQAG\)](/s/profile/0050X000008AC1uQAG) (Customer)
3 years ago

1 limitations and stupid code.

I think the initial goal of the entire library was just to demonstrate how the hardware works, but then many users started to make real use, which means trying to implement the use of multiple hardware modules together and sharing the processing, either by timers or operating system (RTOS etc). So to be a didactic code full of errors. But the good news is that they are looking to solve the problems.

Like Reply 1 like


 [EmbDeveloper \(/s/profile/0053W000000ogb0QAA\)](/s/profile/0053W000000ogb0QAA) (Customer)
Edited December 6, 2021 at 12:45 PM

Hello [Piranha \(https://community.st.com/s/profile/0050X0000088kkUQAQ\)](https://community.st.com/s/profile/0050X0000088kkUQAQ) (Community Member)

Three years have passed, and on the ST website an example of a workable code for NUCLEO-F767ZI for lwIP and IAP based on it did not appear.

Could you please share the code itself?

Like Reply


 [Moe \(/s/profile/0053W000002o6PeQAI\)](/s/profile/0053W000002o6PeQAI) (Customer)
8 months ago

1 I will not provide source code as a whole

I must say that I find your attitude really antisocial. You provide the whole binary but can't provide the source code for it? Are you serious? What logical reason is there for that? Didn't you use the internet or forum posts from other users to make your source code work?

The principle of forums is "take and give". If everyone would think like you, there would soon be no more Internet.

Like Reply 1 like

 [kold \(/s/profile/0050X00000899t2QAA\)](/s/profile/0050X00000899t2QAA) (Customer)
Edited April 24, 2019 at 6:26 AM


A series of guides sounds great.

I have tested the NUCLEO-F767ZI and as reported earlier i runs very good. Once in a while lwip puts out a debug message on the console, but it is still running without any problems, so i might not be anything important. The debug messages:

```
1 0:04:29,877 [DRV_ETH_lwIP] 802.3; RDES0: 0x00400300 RDES4: 0x00000042
2 0:08:28,990 [DRV_ETH_lwIP] 802.3; RDES0: 0x00400300 RDES4: 0x00000042
3 0:10:00,535 [DRV_ETH_lwIP] 802.3; RDES0: 0x00400300 RDES4: 0x00000042
4 0:15:20,951 [DRV_ETH_lwIP] 802.3; RDES0: 0x00400300 RDES4: 0x00000042
```

I tried running the code on a NUCLEO-F746ZG but without any luck. I do not get any output on the serial and none of the leds are turning on.

Like Reply

 [Piranha \(/s/profile/0050X0000088kkUQAQ\)](/s/profile/0050X0000088kkUQAQ) (Customer)
Edited May 11, 2019 at 6:10 AM

I've updated this demonstration firmware and description. Apart from small improvements and fixes, the new version has some substantial improvements:


- Rewritten transmit buffer management.
- Enabled D-cache.
- NUCLEO-F767ZI build made to be compatible with NUCLEO-F746ZG/F756ZG and renamed to NUCLEO-144.

All of these cumulatively reduced CPU load from previous 38,4 % to 18,7 % at the same speed of 94,9 Mbps. It should be noted that 94,9 Mbps is a theoretical maximum for TCP connection over Fast Ethernet in a typical network under ideal conditions. In everyday situation the speed can be little lower, often around 94 Mbps.

P.S. As I have only NUCLEO-F767ZI, someone ([@kold \(/s/profile/0050X00000899t2QAA\)](/s/profile/0050X00000899t2QAA) (Customer) ?) still could test once more NUCLEO-144 build on those added boards.

☺


Like Reply

 [SToma \(/s/profile/0050X0000088evbQAA\)](/s/profile/0050X0000088evbQAA) (Customer)
4 years ago

Hello, [@Piranha \(/s/profile/0050X0000088kkUQAQ\)](/s/profile/0050X0000088kkUQAQ) (Customer),

thank you. It's amazing. Tested with NUCLEO-F746ZG, and it's stable at 95.1 Mbits/sec. Can't wait for sources and comments.

Like Reply

 [ranran \(/s/profile/0050X000007vn6EQAQ\)](/s/profile/0050X000007vn6EQAQ) (Customer)
4 years ago


Hi,

Did you tested it with ethernet test equipment ?

I don't think iperf is reliable as ethernet test equipment.

Thanks

Like Reply

 [SToma \(/s/profile/0050X0000088evbQAA\)](/s/profile/0050X0000088evbQAA) (Customer)


4 years ago

Hi,

no, only with iperf. Measuring with with iperf is quite good for me. But it's irrelevant, because [@Piranha \(/s/profile/0050X0000088kkUQAQ\)](/s/profile/0050X0000088kkUQAQ) (Customer) do not release sources (or better - comments what was changed against ST sources).

Slavo T.

Like Reply 1 like

 [ranran \(/s/profile/0050X000007vn6EQAQ\)](/s/profile/0050X000007vn6EQAQ) (Customer)


4 years ago

Hi [stomascik1.5342336854365312E12 \(https://community.st.com/s/profile/0050X0000088evbQAA\)](https://community.st.com/s/profile/0050X0000088evbQAA),

According to our equipment measurement, we see lost files and error (I did not try it with Piranha binary yet, but I am quite sure I will still see lost with the ethernet equipment), If source or comments are not released, than how we can use this information ? Is it for sale (not that we plan to buy it...)?

Thanks


Like Reply 2 likes

 [kold \(/s/profile/0050X00000899t2QAA\)](/s/profile/0050X00000899t2QAA) (Customer)

4 years ago

I can confirm that the Nucleo-144 build works on the NUCLEO-F746ZG with a speed of 94.9 Mbps. The cpu loads is 27.46 %.

Like Reply

 [ranran \(/s/profile/0050X000007vn6EQAQ\)](/s/profile/0050X000007vn6EQAQ) (Customer)


4 years ago

Hi,

How did you check cpu load ?

Thanks


Like Reply

 [anotherandrew \(/s/profile/0050X000007vo2pQAA\)](/s/profile/0050X000007vo2pQAA) (Customer)

4 years ago

with an RTOS it's pretty easy; set a GPIO pin in the idle task and clear it in the RTOS' task switch routine.

Like Reply

 [elmoood \(/s/profile/0050X000007vvCuQAI\)](/s/profile/0050X000007vvCuQAI) (Customer)

3 years ago

How did you get this working? I have the same dev board and have been fighting for weeks to get things working properly. I tried integrating the H7 code posted by alister, but I just can't get it to even compile. Does anyone have a step by step guide or list of notes on how to improve the stock cube MX code to actually work properly? I'm really upset with ST for shipping such lousy examples. All the claims of amazing performance seem to just dangle an unobtainable carrot... I'm no newb in MCU coding, but I'm pretty lost when it comes to patching / improving whatever has to be improved. I'm not even sure what I am supposed to know before getting into this...

Like Reply

 [Seyed \(/s/profile/0053W000001odZAQAY\)](/s/profile/0053W000001odZAQAY) (Customer)


2 years ago

Hi there kold

How did you tested the ethernet speed?

I'm trying to test UDP speed on a custom borad and looking for a standard way to measure the speed.

Like Reply


 [kold \(/s/profile/0050X00000899t2QAA\)](/s/profile/0050X00000899t2QAA) (Customer)

2 years ago

Hi Seyed

Well it is a long time since I tested it, but if I remember correctly I used iperf for testing and it must have been TCP speed I was testing.


Like Reply

 [Seyed \(/s/profile/0053W000001odZAQAY\)](/s/profile/0053W000001odZAQAY) (Customer)

2 years ago

Gonna check it out, thanks

Like Reply


 [zoromgerg \(/s/profile/0050X000007vtrCQAQ\)](/s/profile/0050X000007vtrCQAQ) (Customer)

4 years ago

Nice, so FW is able to acknowledge fast the input data transfer from client. That might be possible even there was RX error. As FW sources are closed, would be good at least to return checksum of received data via virtual serial port or http output.

How about transmitting some data back to client and doing measurements for that?

Like Reply


 [Piranha \(/s/profile/0050X0000088kkUQAQ\)](/s/profile/0050X0000088kkUQAQ) (Customer)

4 years ago

What You are saying is true for UDP and raw Ethernet, but not for TCP. TCP in itself provides reliable, ordered, and error-checked delivery of a stream of bytes. Therefore, if one end is able to send data continuously, it implies that the other end is receiving all of that data unaltered. Checksums and acknowledgements are managed by TCP part of IP stack.

I have a desire myself to test transmit and full-duplex performance also. In fact it's almost ready. I still have to complete and test some details and then I'll publish new version of this firmware.

Like Reply


 [zoromgerg \(/s/profile/0050X000007vtrCQAQ\)](/s/profile/0050X000007vtrCQAQ) (Customer)

4 years ago

I've started my first comment in wrong manner. Your results look great. I just was thinking that proposed here setup can't confirm that data actually are available for application layer. Sure, I can look skeptical (or/and jealous) about described results, but there is no sufficient way to ensure proper data delivery since drivers might implement some shortcuts in order to provide expected acknowledgements for client w/o taking care about real data. Especially when TCP checksums aren't used. That is why I've proposed some additional output with result of some operation upon RX data as sufficient prove.

I wish you good time with full-duplex implementation.

Like Reply

 [rromano001 \(/s/profile/0050X0000088mmiQAA\)](/s/profile/0050X0000088mmiQAA) (Customer)

Edited July 22, 2021 at 8:07 PM

Hello Piranha, reading all your post about I got a doubt about firmware and bug I discovered about auto negotiation, is original firmware able to do full duplex or setting duplex/speed by wrong register was a choice or not understood behavior?

Please, can you compile for stm32F407VE or can I send you a board to test on?


this part is broken too but seems making no difference from working F429Z to unresponsive F407VE

F429Z nucleo doesn't show differences with broken or patched code.

```
1 /* Read the result of the auto-negotiation */
2     HAL_ETH_ReadPHYRegister(&heth, PHY_SR, &regvalue); <<<-- //Wrong register forever return 0x41 regardless of negotiated speed
   and duplex
3
4 // patch to read right register 0x1f
5     HAL_ETH_ReadPHYRegister(&heth, 0x1f, &regvalue);
6
7     /* Configure the MAC with the Duplex Mode fixed by the auto-negotiation process */
8     if((regvalue & /*PHY_DUPLEX_STATUS*/ 0x10) != (uint32_t)RESET) <<<-- // what is RESET and why use non sense values??? Right
   mask is 0x10
9     {
10        /* Set Ethernet duplex mode to Full-duplex following the auto-negotiation */
11        heth.Init.DuplexMode = ETH_MODE_FULLDUPLEX;
12    }
13    else
14    {
15        /* Set Ethernet duplex mode to Half-duplex following the auto-negotiation */
16        heth.Init.DuplexMode = ETH_MODE_HALFDUPLEX;
17    }
18    /* Configure the MAC with the speed fixed by the auto-negotiation process */
19    if(regvalue & PHY_SPEED_STATUS) <<-- // this is set to 2, must be 4 or 8, 10base T require value 4, best is to use mask 0x0c
   then test 4 to set 10BT and 8 for 100BT
20    {
21        /* Set Ethernet speed to 10M following the auto-negotiation */
22        heth.Init.Speed = ETH_SPEED_10M;
23    }
24    else
25    {
26        /* Set Ethernet speed to 100M following the auto-negotiation */
27        heth.Init.Speed = ETH_SPEED_100M;
28    }
29    }
30    else /* AutoNegotiation Disable */
31    {
32        error :
33        /* Check parameters */
34        assert_param(IS_ETH_SPEED(heth.Init.Speed));
35        assert_param(IS_ETH_DUPLEX_MODE(heth.Init.DuplexMode));
36
37        /* Set MAC Speed and Duplex Mode to PHY */
38        HAL_ETH_WritePHYRegister(&heth, PHY_BCR, ((uint16_t)(heth.Init.DuplexMode >> 3) |
39                                (uint16_t)(heth.Init.Speed >> 1)));
40    }
```

Thank you for all hint and information, so sorry but tomorrow I call FAE to report this. Component shortage is not the only issue. ☹️

Like Reply

 [Piranha \(/s/profile/0050X0000088kkUQAQ\)](/s/profile/0050X0000088kkUQAQ) (Customer)


4 years ago

A new version is here with a following improvements:

- Replaced my basic lperf2 server implementation with the one provided together with lwIP.
- Reduced CPU load by approximately 6 % relative to previous results.

The new lperf2 server supports receive and transmit testing in half and full duplex modes. Description has been updated accordingly.


Like Reply 1 like

 [zoromgerg \(/s/profile/0050X000007vtrCQAQ\)](/s/profile/0050X000007vtrCQAQ) (Customer)

4 years ago

Great job!


Like Reply

 [Bryan \(/s/profile/0050X000007RTf8QAG\)](/s/profile/0050X000007RTf8QAG) (Customer)

4 years ago

I would love to see a guide or to see your source! I've had endless frustrations trying to bring up ethernet+lwIP on stm32 chips, and would really like to see something that actually works reliably.


Like Reply 1 like

 [valik mahrov \(/s/profile/0050X000007vu2LQAQ\)](/s/profile/0050X000007vu2LQAQ) (Customer)

4 years ago

Can your program work more than 1 user?


Like Reply

 [Piranha \(/s/profile/0050X000088kkUQAQ\)](/s/profile/0050X000088kkUQAQ) (Customer)

4 years ago

What exactly do You mean by "user"?


Like Reply

 [valik mahrov \(/s/profile/0050X000007vu2LQAQ\)](/s/profile/0050X000007vu2LQAQ) (Customer)

4 years ago

TCP-clients

Like Reply


 [Piranha \(/s/profile/0050X000088kkUQAQ\)](/s/profile/0050X000088kkUQAQ) (Customer)

4 years ago

Iperf2 server is the one, that comes together with lwIP package, and yes, it supports multiple connections simultaneously. You can run multiple Iperf2 clients and test for Yourself.

HTTP server is built by me and is very trivial. It doesn't limit connections, but, as it renders page and after that immediately closes connection, the question becomes kind of irrelevant.

Like Reply


 [anotherandrew \(/s/profile/0050X000007vo2pQAA\)](/s/profile/0050X000007vo2pQAA) (Customer)

4 years ago

This is fantastic, but without source it is nothing more than grandstanding. We can't recreate your success nor leverage it in our own designs. I'm looking forward to your future topics.

Would you be willing to provide a quick bullet list of any tricks/tips/bugs while you work on the more involved/complete topics?

Like Reply 3 likes

 [Piranha \(/s/profile/0050X000088kkUQAQ\)](/s/profile/0050X000088kkUQAQ) (Customer)


4 years ago

I do agree. OK, here are bullet list of main problems particularly related to Ethernet and lwIP, starting from more serious ones:

- Rx buffer position and size not aligned to cache line size in projects, where cache management functions are used.
- Missing or wrong cache management function (clean and invalidate) use in some places.
- Not respecting lwIP multi-threading requirements in lwIP driver and applications. Not protected DMA descriptor management and *pbuf_custom* freeing. Using non-thread-safe functions without core locking.
- Rx buffer overrun possible if frames come in faster than receiving thread processes them or Tx buffer overrun possible if output frames are queued faster than hardware can transmit.
- DMB memory barriers missing before OWN bit setting and before checking/resuming DMA processing.

At least some more detailed topics about these problems are starting in this or next week.

Like Reply

 [anotherandrew \(/s/profile/0050X000007vo2pQAA\)](/s/profile/0050X000007vo2pQAA) (Customer)


4 years ago

Actually I just got bit by the DMA issue. F756 with RMII ethernet an FPGA hanging off of SPI5. Ethernet was up and working reasonably well, but when I went to load the ~120kB bitstream to the FPGA ethernet would go "deaf". I could see packets the STM transmitted but it saw nothing coming in. Don't init the FPGA and it's fine. Init and it's deaf. Completely reproducible.

I spent an entire day trying to see what was going on (suspected that I was perhaps killing the GPIO AF config for the RX pins) but nothing. Out of frustration I started disabling entire subsystems to narrow the problem down. Turns out that when I did not enable D cache I could load the FPGA and still have ethernet. I had a weird problem where it seems that every OTHER packet would finally cause LwIP to process both, but what a frustrating day.

Looked into it a bit more and yes -- no D cache management at all, and this is v1.14 of the STM32F7 HAL drivers/middleware!

Like Reply

 [zoromgerg \(/s/profile/0050X000007vtrCQAQ\)](/s/profile/0050X000007vtrCQAQ) (Customer)

4 years ago

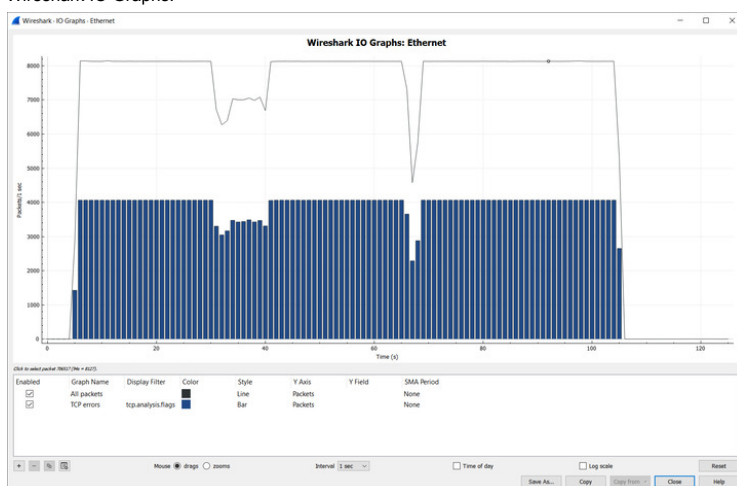
Interesting, am I only who experience the following non-stable bandwidth issue?

HW: NUCLEO-F767ZI

SW: latest proposed by the topic

Test sequence: `iperf -c <ip_address> -t 100`


Wireshark IO Graphs:



This observation (drops in speed) has been visible for me so far on each test session with 100s interval. The board is connected directly to the PC. Zero packets are lost during test communication. Can someone reproduce this issue?

As additional info, I already can trace the root cause for my case up to Ethernet PHY behavior. More info after someone can confirm this observation. And I've already checked that ST Ethernet HAL doesn't behave better for this bandwidth issue.

Like Reply

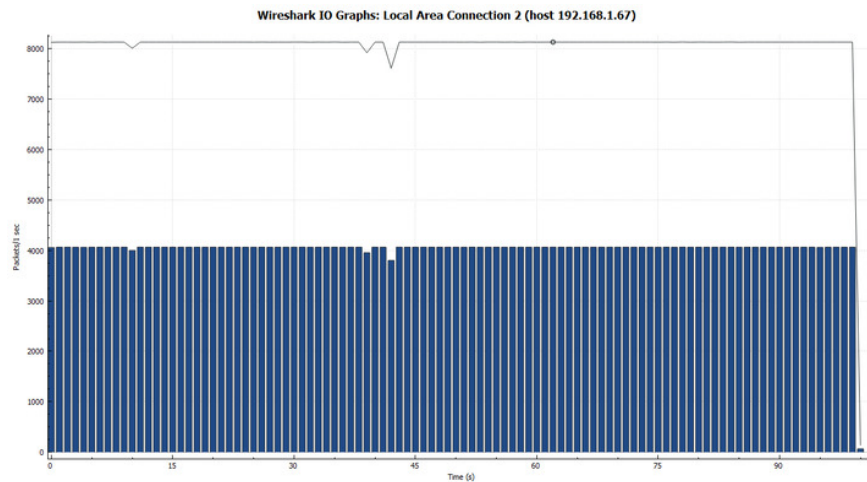
 [Piranha \(/s/profile/0050X000088kkUQAQ\)](#) (Customer)

Edited August 14, 2019 at 7:54 AM

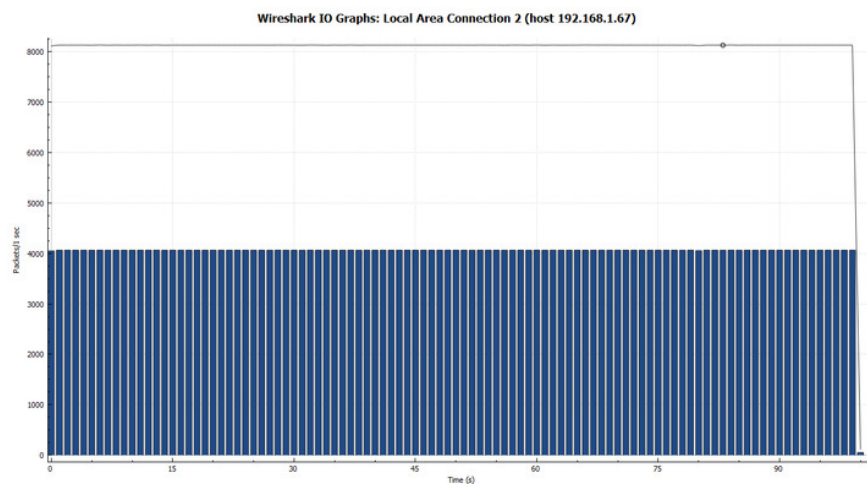
Unfortunately I can't confirm this behavior.

Setup: NUCLEO-F767ZI <=> MikroTik RB951G-2HnD <=> Athlon II X2 270, Windows 7 x64.

Worst capture with previous firmware version:



Best capture with updated firmware version:




While typically both versions being somewhere in between these two examples. I would check these points:

1. Other software interfering with network.
2. Turning off real-time protection or even uninstalling antivirus and firewall software.
3. Scanning for malware (for example with Malwarebytes).
4. Changing Ethernet cables.
5. Updating all OS drivers (not only for network interfaces).

P.S. Answering Your previous proposal of data checksum - that would not be Iperf2 protocol, which means that I would need to develop not only a server for MCU, but also a client software for a PC. Honestly, I'm not interested in that. Also that would add some additional CPU load, which would not be network related, but my intention here is to show solely network performance, not application. ☺

Like Reply


 [zoromgerg \(/s/profile/0050X00007vtrCQAQ\)](#) (Customer)

4 years ago

Thank you for the checking. After additional tests with different network setup (router now is in-between instead of direct connection) gives result w/o any speed drops.

In regards to my proposal to add data checksum. While your firmware is closed and is sending (or receiving) only zeroes, there is no guarantee that you don't provide any shortcut on low level which frees firmware from doing additional data coping. And removing of that extra coping might give better performance measurement result.


Like Reply

 [anotherandrew \(/s/profile/0050X00007vo2pQAA\)](#) (Customer)

4 years ago

@piranha -- it's been a couple of months now -- any plan to release the source? I'd like to replicate this, or at least test your code with my RMII-connected F756 which seems to be having trouble receiving. It doesn't appear to be a software issue, as I've been able to determine that I'm getting interrupts for only approximately half the packets received. Transmitting is not an issue, just receiving.

Like Reply


 [Piranha \(/s/profile/0050X000088kkUQAQ\)](#) (Customer)

Edited August 14, 2019 at 8:32 AM

No source code, but detailed "to the point" topics" are just around the corner... ☺

About interrupts - that sounds like normal processing. Interrupts have a flag. Therefore receive interrupt really means "at least one frame has been received". When it happens, Ethernet processing task must loop through all new received frames (descriptor OWN bit cleared) and pass those to IP stack. Only after that has been done completely, it can go for waiting on a semaphore or better signals/events/notifications (it's silly that ST doesn't use those).

Like Reply

 [anotherandrew \(/s/profile/0050X000007vo2pQAA\)](/s/profile/0050X000007vo2pQAA) (Customer)
4 years ago

I understand that it's not "interrupt per frame" -- the IRQ handler calls the HAL_ETH_IRQHandler() which calls HAL_ETH_RxCpltCallback() if it's an interrupt caused by a raised "received frame" flag. The LwIP callback in ethernetif.c releases the LwIP "got a frame" semaphore, and the ethernetif_input() thread that is waiting on the semaphore repeatedly calls low_level_input() to pull in frames. low_level_input() calls HAL_ETH_GetReceivedFrame_IT() and processes the RxFrameInfos struct to build up a pbuf for the received packet (one frame per call to HAL_ETH_GetReceivedFrame_IT()). ethernetif_input() calls low_level_input() in a loop until low_level_input() returns NULL, meaning no more frames.


So as you see, I am already pulling in all available frames whenever the IRQ fires. I've added counters for "got an interrupt", "RxCplt callback called", "HAL_ETH_GetReceived didn't return HAL_OK" and "rxFrameInfos.length is zero". I then view these either through SWO or by updating an attached LCD in another thread.

The network is quiet and the only traffic is my once-per-second ICMP echo (ping) packets. With that kind of network traffic, I should be getting an interrupt every second. I am not; I am getting an interrupt roughly once every other second, with only one packet processed. SOMETIMES I get an interrupt every second, sometimes not for several seconds... it's my conclusion that the STM32 MAC is simply not seeing the packets coming in.

I've also adjusted HAL_MACDMAConfig() (called from HAL_ETH_Init()) so that I have turned on/enabled "receive all", "promiscuous mode" "forward errored frames" and "forward undersized good frames" in an attempt to see if it's getting malformed/bad packets. I've also Disabled "drop TCP/IP checksum errored frames". Further, I've enabled abnormal interrupts (all the receive ones: receive FIFO overflow, receive buffer unavailable, receive process stopped, receive watchdog timeout and fatal bus error) and added counters there from the HAL_ETH_ErrorCallback() -- no errors ever, just missed frames.


Finally, I also capture the DMAMFBOCR register; it's one of those "read clears all bits" registers so I latch it if it's ever nonzero. It's always zero UNLESS I stop the running program with the debugger for more than 4-5 seconds. THEN BOCR shows missed frames which makes perfect sense; the DMA operates even when the CPU is halted but runs out of empty descriptors after a few seconds.

Like Reply 1 like

 [anotherandrew \(/s/profile/0050X000007vo2pQAA\)](/s/profile/0050X000007vo2pQAA) (Customer)
4 years ago


I'm wondering if you'd mind sharing (just in point form) where specifically you've had to address "Not respecting lwIP multi-threading requirements in lwIP driver and applications. Not protected DMA descriptor management and *pbuf_custom* freeing. Using non-thread-safe functions without core locking." I've taken a quick look but not being super familiar with the soft bits of LwIP, it's hard to notice the problem areas.

Like Reply

 [Piranha \(/s/profile/0050X0000088kkUQAQ\)](/s/profile/0050X0000088kkUQAQ) (Customer)
4 years ago


Good description! I suggest that You make a separate topic with this description and let's continue discussing this particular issue there, because otherwise this topic will quickly become unreadable within this "ingenious" forum engine. I'll help and possibly that topic can also become a bug report for ST. Also post a link here for anyone interested in continuation.

Like Reply

 [anotherandrew \(/s/profile/0050X000007vo2pQAA\)](/s/profile/0050X000007vo2pQAA) (Customer)
4 years ago


Actually, it appears that my issue is not in the STM32 code nor the HAL; I took the codebase and with very, very minor changes (pin mapping), flashed it to a Nucleo-F756ZG board. The network came up and there is no packet loss. The problem is thus likely to lie either in timing closure to the FPGA or in the HDL of the FPGA itself. Pity, debug tools are so much nicer in software land.

Like Reply

 [anotherandrew \(/s/profile/0050X000007vo2pQAA\)](/s/profile/0050X000007vo2pQAA) (Customer)
4 years ago


Got it resolved -- it was an issue in my HDL code, in my RMII-to-MII bridge. Getting 0% packet loss now over many tens of thousands of packets. Back to the real work now!

Like Reply 1 like

 [Petew \(/s/profile/0050X000009zC23QAE\)](/s/profile/0050X000009zC23QAE) (Customer)
Edited March 10, 2020 at 7:52 PM

This is an excellent post

Like Reply

 [Piranha \(/s/profile/0050X0000088kkUQAQ\)](/s/profile/0050X0000088kkUQAQ) (Customer)
Edited August 14, 2019 at 8:37 AM

Hi, everyone!


I've released a new version with minor performance and other minor improvements. Still it managed to lower CPU load by 0,6/0,4 percentage points (absolute percentage) under Rx/Tx loads respectively. 😊

Sorry for delays with the problem topics - there are still one small issue I have to understand fully myself to write a topic which explains it to others. But meanwhile You can already read/fix/implement this:

<https://community.st.com/s/question/0D50X0000B2AG7FSQW/ethernet-send-complete-interrupt-not-triggered-in-stm32f7>


<https://community.st.com/s/question/0D50X0000B2AG7FSQW/ethernet-send-complete-interrupt-not-triggered-in-stm32f7>

Like Reply 1 like

 [valik mahrov \(/s/profile/0050X000007vu2LQAQ\)](/s/profile/0050X000007vu2LQAQ) (Customer)
Edited August 23, 2019 at 1:38 PM


Hi! Has anyone made an HTTP server with an unlimited number of clients on STM32?

Like Reply

 [anotherandrew \(/s/profile/0050X000007vo2pQAA\)](/s/profile/0050X000007vo2pQAA) (Customer)
4 years ago

an HTTP server has no concept of "clients" -- you connect to a port, send a request, get a response. There will be practical (RAM) limitations about how many simultaneous connections can be handled, but this isn't clients so much as connections.


Like Reply

 [valik mahrov \(/s/profile/0050X000007vu2LQAQ\)](/s/profile/0050X000007vu2LQAQ) (Customer)

4 years ago

I have this kind of problem <http://lwip.100.n7.nabble.com/Incomprehensible-behavior-with-the-LWIP-library-on-STM32-td34603.html> (<http://lwip.100.n7.nabble.com/Incomprehensible-behavior-with-the-LWIP-library-on-STM32-td34603.html>) , but I don't know how to solve it. I have been solving it myself for 6 months, but without result.

Like Reply

 [ranran \(/s/profile/0050X000007vn6EQAQ\)](/s/profile/0050X000007vn6EQAQ) (Customer)

Edited November 29, 2019 at 11:02 AM

Hello [Piranha \(https://community.st.com/s/profile/0050X0000088kkUQAQ\)](https://community.st.com/s/profile/0050X0000088kkUQAQ) (Community Member),

Thank you for sharing the binary of the demonstration.

I see that you tested it with iperf from another PC host.


We have tested performance using Ethernet test equipment.

1. I am not sure that PC iperf can reach ieee 802.3 required performance. The test equipment generate packets and interval between packets exactly according to ieee 802.3 requirement (the interval timing is accurate and we can not change it even if we want to).
2. What is the packet length in your tests ? Did you try packets below ~150 bytes ?
3. How do you check cpu load/utilization ?
4. Is it done in baremetal or RTOS (if it is baremetal it won't help me because I use must use RTOS) ?

Thank you!

ranran

Like Reply 1 like


 [Piranha \(/s/profile/0050X0000088kkUQAQ\)](/s/profile/0050X0000088kkUQAQ) (Customer)

Edited December 3, 2019 at 10:40 AM

1. I don't know what your test equipment is or does, but it seems that it is meant to test hardware implementation, which is useful, when developing Ethernet MAC in FPGA or ASIC. Ethernet peripheral in STM32 is already implemented as conforming to IEEE 802.3 and one also can't modify Ethernet frame timings and other hardwired parameters.
2. The TCP protocol doesn't expose control over frames to user API. But, to reach the maximum 94,9 Mbps on Fast Ethernet as is the case with this demonstration, there is only one option - frames with a payload size of MTU (1500 bytes) must be used. But the code of course works with any IEEE 802.3 legal frame size with 46-1500 bytes payload.
3. I've implemented code based on *DWT->CYCCNT*. It counts the cycles, which CPU spends in sleep mode and total cycles. I accumulate both until total cycles reach the amount of *SystemCoreClock* (one second) and then do subtract sleep cycles from total cycles. The result is cycles of the previous second, which was spent on all processing, including interrupts, context switches and stalls.
4. The topic description already contains the answer - this demonstration uses FreeRTOS, but doesn't have any ST's code. And, by the way, if one has a decent cooperative scheduler and software timer implementation, then it is even easier to implement networking and other things on a "bare-metal".

P.S. I'll give other related comments in your topic. Link for people interested: <https://community.st.com/s/question/0D50X0000BNu2BFSQZ> (<https://community.st.com/s/question/0D50X0000BNu2BFSQZ>)

Like Reply

 [ranran \(/s/profile/0050X000007vn6EQAQ\)](/s/profile/0050X000007vn6EQAQ) (Customer)

3 years ago


I also used the iperf2 for testing performance and managed to get 100Gb/sec.

Yet, on testing the exact same embedded application with Ethernet equipment we see errors.

It means that iperf2 is not reliable for Ethernet performance testing !

Thanks

Like Reply

 [TBuls \(/s/profile/0050X000008A3CFA0\)](/s/profile/0050X000008A3CFA0) (Customer)


Edited January 30, 2020 at 3:07 PM

This is great news, as i have been spending some time now trying to get stuff running stable. However....

Is there any chance of you sharing or selling the sources for us to benefit from or are you just wanting to make ST look bad? What is the holdup on these series of articles of yours? Up to now you have not been much more helpful than ST to be honest...

[edit] The latter remark is not true, running through the referenced topics i see you add more detailed bugfixes..


Like Reply 1 like

 [ranran \(/s/profile/0050X000007vn6EQAQ\)](/s/profile/0050X000007vn6EQAQ) (Customer)

3 years ago

Please also note that the tool you used within these articles to see if there is any improved performance is iperf2. As I said earlier, when I tested it with Ethernet traffic/analyzer it showed lower performance results compared with iperf2.

Like Reply


 [ASánchezRoncero \(/s/profile/0050X000009yuUOQAY\)](/s/profile/0050X000009yuUOQAY) (Customer)

3 years ago

Hi, could you give a brief explanation of how to integrate de full-duplex tcp with FreeRTOS?? At this point, I succesfully implemented it in just 1 dir, but having problems with the code generated by ST for the dúplex mode. (Also F746ZG)

Thank you!!

Like Reply

 [anotherandrew \(/s/profile/0050X000007vo2pQAA\)](/s/profile/0050X000007vo2pQAA) (Customer)

3 years ago

I'm just updating this thread to point out a followup thread where [@Piranha \(/s/profile/0050X0000088kkUQAQ\)](/s/profile/0050X0000088kkUQAQ) (Customer) has gathered all the information and the solutions:

<https://community.st.com/s/question/0D50X0000B0ffhnSQB/how-to-make-ethernet-and-lwip-working-on-stm32>

<https://community.st.com/s/question/0D50X0000B0ffhnSQB/how-to-make-ethernet-and-lwip-working-on-stm32>

Like Reply

[LCE \(/s/profile/0053W000001hfJKQAY\)](#) (Customer)

Edited September 6, 2022 at 4:17 PM

[@Piranha \(/s/profile/0050X0000088kkUQAQ\)](#) (Customer)

How did you achieve this TCP speed on the transmit side concerning lwIP?

- 1) Did you modify any lwIP functions?
- 2) Did you simply use the provided `tcp_write()` function and let lwIP handle the rest?
- 3) What are important optimization settings (e.g. in lwIP's `opt.h`) ?

I am mostly curious about 2) `tcp_write()`, because I'm thinking about giving complete buffers (little smaller than 1500 bytes) and sending these directly with a forced `tcp_output()` directly after, or would using PCB flag `TF_NODELAY` to disable Nagle algorithm be good enough?

I think sending complete data buffers will simplify the freeing of these after TCP's ACK, compared to TCP packets composed of multiple data buffers.

Thanks in advance, I hope you have a minute for a reply.

Like Reply

[Piranha \(/s/profile/0050X0000088kkUQAQ\)](#) (Customer)

Edited September 7, 2022 at 12:40 AM

None of the used external libraries were modified. Not a single byte! Iperf2 server is just the "[lwiperf \(https://github.com/lwip-tcpip/lwip/blob/master/src/apps/lwiperf/lwiperf.c\)](https://github.com/lwip-tcpip/lwip/blob/master/src/apps/lwiperf/lwiperf.c)" coming with lwIP and that also was not modified.

With TCP I've made my own Telnet server. And indeed for it I'm disabling the Nagle's algorithm and calling `tcp_output()` after every `tcp_write()`. But that is Telnet - the latency is important for it, not the data rate.

Not much of a magic in configuration either:

```
1 // ETH lwIP driver
2 #define DRV_ETH_LWIP_BUF_RX_SIZE      1536
3 #define DRV_ETH_LWIP_BUF_RX_NUM      16
4 #define DRV_ETH_LWIP_BUF_TX_NUM      48
5
6 // lwipopts.h
7 #define MEMP_NUM_PBUF                DRV_ETH_LWIP_BUF_TX_NUM
8 #define MEMP_NUM_TCP_SEG            TCP_SND_QUEUELEN
9 #define TCP_WND                      (8 * TCP_MSS)
10 #define TCP_MSS                     1460
11 #define TCP_SND_BUF                 TCP_WND
```

Another important point - I'm using [custom memory pools \(https://lwip.fandom.com/wiki/Custom_memory_pools\)](https://lwip.fandom.com/wiki/Custom_memory_pools) instead of a heap.

Like Reply 2 likes

[LCE \(/s/profile/0053W000001hfJKQAY\)](#) (Customer)

9 months ago

Thank you very much for your reply!

Latency might also be an issue for me, so one more reason it will be `tcp_output()` after every `tcp_write()`.

I will also have a look at the custom memory pools.

Like Reply

Log In to Answer

ST Community

[Q&A \(https://community.st.com/s/topiccatalog\)](https://community.st.com/s/topiccatalog)

[Groups \(https://community.st.com/s/group/CollaborationGroup/00Bb0000003cRLJEA2\)](https://community.st.com/s/group/CollaborationGroup/00Bb0000003cRLJEA2)

[Projects \(https://community.st.com/s/project/Project_c/00B3W000001Ee8kUAC\)](https://community.st.com/s/project/Project_c/00B3W000001Ee8kUAC)

[Ideas \(https://community.st.com/s/ideazone\)](https://community.st.com/s/ideazone)

[Knowledgebase \(https://community.st.com/s/toparticles\)](https://community.st.com/s/toparticles)

[Academy \(https://community.st.com/s/learning-catalogs\)](https://community.st.com/s/learning-catalogs)

[Help \(https://community.st.com/s/community-overview\)](https://community.st.com/s/community-overview)

From st.com

[st.com Home \(https://www.st.com/\)](https://www.st.com/)

[Solutions \(https://www.st.com/content/st_com/en/products/solutions-reference-designs.html\)](https://www.st.com/content/st_com/en/products/solutions-reference-designs.html)

[Careers \(https://www.st.com/content/st_com/en/about/careers.html/\)](https://www.st.com/content/st_com/en/about/careers.html/)

[Buy From the eStore \(https://estore.st.com/en/\)](https://estore.st.com/en/)

[Wiki \(https://www.st.com/content/st_com/en/wiki/wiki-portal.html/\)](https://www.st.com/content/st_com/en/wiki/wiki-portal.html/)

[ST Blog \(https://blog.st.com/\)](https://blog.st.com/)

Connect with ST

[Submit Support Request \(https://my.st.com/ols\)](https://my.st.com/ols)

[Contact ST Offices \(https://www.st.com/content/st_com/en/contact-us.html\)](https://www.st.com/content/st_com/en/contact-us.html)

[Find Sales Offices & Distributors \(https://www.st.com/content/st_com/en/contact-us.html\)](https://www.st.com/content/st_com/en/contact-us.html)

[Media Center \(https://newsroom.st.com/\)](https://newsroom.st.com/)

[Events \(https://www.st.com/content/st_com/en/about/events/upcoming-events-and-technical-seminars.html\)](https://www.st.com/content/st_com/en/about/events/upcoming-events-and-technical-seminars.html)

Privacy

[Privacy Portal \(https://www.st.com/content/st_com/en/common/privacy-portal.html\)](https://www.st.com/content/st_com/en/common/privacy-portal.html)

Follow Us

[!\[\]\(a870788d6ed9b8fd294b7654a8c8526b_img.jpg\) \(https://twitter.com/st_world\)](https://twitter.com/st_world) [!\[\]\(18065afa4ef6662bca9f3f6088f7de30_img.jpg\) \(https://www.facebook.com/STMicroelectronics.NV\)](https://www.facebook.com/STMicroelectronics.NV) [!\[\]\(b985170eefb48b9b3ef593e79310e8f5_img.jpg\) \(http://www.youtube.com/user/STonlineMedia\)](http://www.youtube.com/user/STonlineMedia) [!\[\]\(65defa7fe6c24be84c2514c965593962_img.jpg\)](https://www.linkedin.com/company/stmicroelectronics)
[!\[\]\(fcdd23ff0d6d447f60ed83fa544a6d33_img.jpg\)](https://www.linkedin.com/company/stmicroelectronics) [!\[\]\(702f22e7ad1b700defb72bcd0f4a3dad_img.jpg\) \(https://www.instagram.com/stmicroelectronics.nv/\)](https://www.instagram.com/stmicroelectronics.nv/)

All rights reserved © 2021 STMicroelectronics | [Terms of use \(https://www.st.com/content/st_com/en/common/terms-of-use.html\)](https://www.st.com/content/st_com/en/common/terms-of-use.html) | [Sales Terms & Conditions \(https://www.st.com/content/cc/resource/legal/legal_agreement/sales_agreement/49/05/8c/68/a0/35/4a/6e/ST%20Terms%20and%20Conditions%20of%20Sales.pdf/files/ST%20Terms%20and%20Conditions%20of%20Sales.pdf\)](https://www.st.com/content/cc/resource/legal/legal_agreement/sales_agreement/49/05/8c/68/a0/35/4a/6e/ST%20Terms%20and%20Conditions%20of%20Sales.pdf/files/ST%20Terms%20and%20Conditions%20of%20Sales.pdf) | [Trademarks \(https://www.st.com/content/st_com/en/about/st_company_information/stmicroelectronics-trademark-list.html\)](https://www.st.com/content/st_com/en/about/st_company_information/stmicroelectronics-trademark-list.html) | [Privacy Portal \(https://www.st.com/content/st_com/en/common/privacy-portal.html\)](https://www.st.com/content/st_com/en/common/privacy-portal.html)